

NECパーソナルコンピュータ

PC-8800シリーズ

NEW

PC-88VA

テクニカル マニュアル

PC-88VA
Technical Manual

監修・システムソフト
BNN第2企画部編

BNN
Only BNN's Network

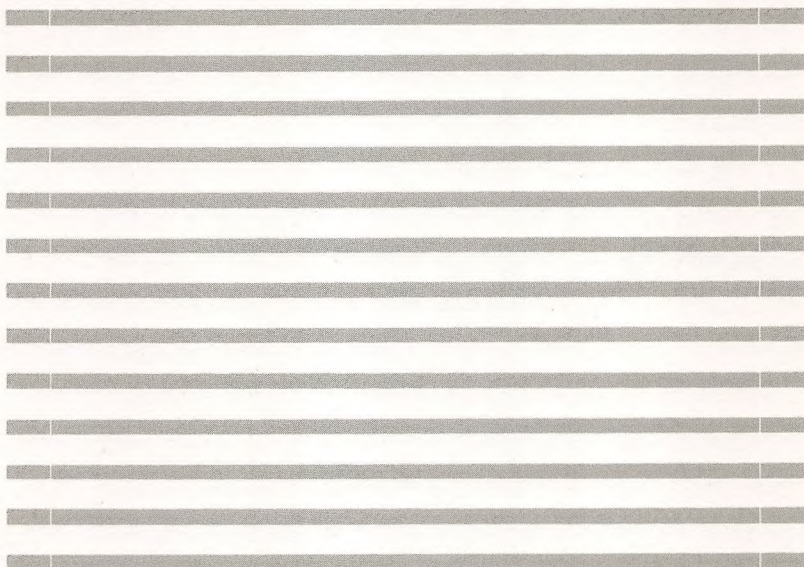
NECパーソナルコンピュータ
PC-8800シリーズ

NEW

PC-88

VA

テクニカルマニュアル



PC-88VA
Technical Manual

監修・システムソフト
BNN第2企画部編

BNN
Bug News Network

はじめに

長いあいだ8ビットパーソナルコンピュータの代表格として多くのユーザに親しまれてきたPC-88シリーズの最上位機種として16ビットのCPUを持つマシンが昭和62年春にリリースされました。「PC-88VA」です。

このマシンは、 μ PD9002 と呼ばれる日本電気が独自に開発した16ビットCPUを搭載しており、PC-88シリーズのラインアップとしては初めての16ビットパソコンとして大きな注目を集めています。その秘められた能力は、たとえば、Z80 エミュレーションによるPC-88シリーズ対応ソフトウェア資産の継承、新開発のDOS(通称、PC-Engine)による多彩なグラフィックス&サウンド表現、高速日本語処理機能などどれをとってもパーソナルユースのパソコン環境として十分な威力を発揮しています。

また、近年より高度な利用が可能になりつつあるパソコン通信への対応、各家庭に普及しているビデオとの連動など、周辺の環境も整備されており、PC-88シリーズのイメージリーダーとして確固たる地位を築いていると言えます。

*

本書の基本編集方針は、PC-88VA上でアプリケーションやペリフェラルをプログラミングあるいは開発する際に重宝するマシンサイド・ディクショナリとして機能させるところにあります。従って、PC-88VAのハードウェア・ソフトウェアのテクニカル情報を詳らかにすることを中心に記述・編集されており、付随する解説は必要最小限にとどめてありますので、本書を活用するにあたっては相応の知識を要します。あらかじめ、御了承ください。

本書の制作については、万全の態勢でのぞんでおりますが、万一誤記、記述もれ等が見受けられたときは御容赦願います。また、記述の内容につきましては、ある特別の条件下では正常に動作しない場合があるかもしれません。その場合につきましては責任を負うことはできませんので、あらかじめ慎重に対応するよう重ねてお願い申し上げます。

目 次

PC-88VAテクニカルマニュアル

第1章 システム概説

7

1.1	ハードウェアブロック図	7
1.2	機能スペック	8
1.3	ソフトウェア動作環境	11
1.4	システム起動プロセス	12
1.5	CPU	12

第2章 マップ

15

2.1	割込みベクタ	15
2.2	I/Oポートマップ	17

第3章 メモリ

47

3.1	メモリモード	47
3.2	メモリレイアウト	47
3.3	メモリコントロール	50
3.4	バックアップメモリ	52
3.5	ウェイト	53
3.6	拡張スロットで拡張できるメモリ	54
3.7	DMAによるメモリアクセス(V3モード)	54

第4章 表示システム

55

4.1	表示システム概要	55
4.2	テキスト表示機能	58
4.3	スプライト表示機能	66
4.4	TSP(テキスト/スプライトプロセッサ)	71

4.5	グラフィック機能	76
4.6	カラーとパレット	103
4.7	その他の機能	108
4.8	文字セット／文字コード／文字フォント	111
4.9	CRT走査モードと表示解像度	117
4.10	ビデオディжитाइズ	121

第5章 その他のハードウェアコントロール 125

5.1	スイッチとシステム動作モード	125
5.2	割込み	127
5.3	DMA機能	132
5.4	タイマ機能	133
5.5	カレンダー時計	135
5.6	キーボードインターフェイス	139
5.7	ディスクインターフェイス	142
5.8	プリンタインターフェイス	148
5.9	RS-232C インターフェイス	148
5.10	サウンド機能	150
5.11	マウスインターフェイス	154
5.12	スキャナインターフェイス	156

第6章 BIOS 159

6.1	フロッピーディスクBIOS	159
6.2	ハードディスクBIOS	173
6.3	キーボードBIOS	183
6.4	テキストBIOS	200
6.5	スプライトBIOS	218
6.6	グラフィック画面制御BIOS	228

6.7 拡張グラフィックBIOS	235
6.8 アニメーションBIOS	252
6.9 プリンタBIOS	259
6.10 コミュニケーションBIOS	263
6.11 サウンドBIOS	273
6.12 マウスBIOS	285
6.13 カレンダー時計BIOS	293
6.14 ファンシーフォントBIOS	296
6.15 スクリーンエディタBIOS	298
6.16 日本語入力フロントプロセッサBIOS	303
6.17 数値演算BIOS	314
6.18 出力専用ポート保存領域	321

第7章 ファイルシステム 323

7.1 システムインターラプトコール	323
7.2 プロセスコントローラ	329
7.3 システムコール	329

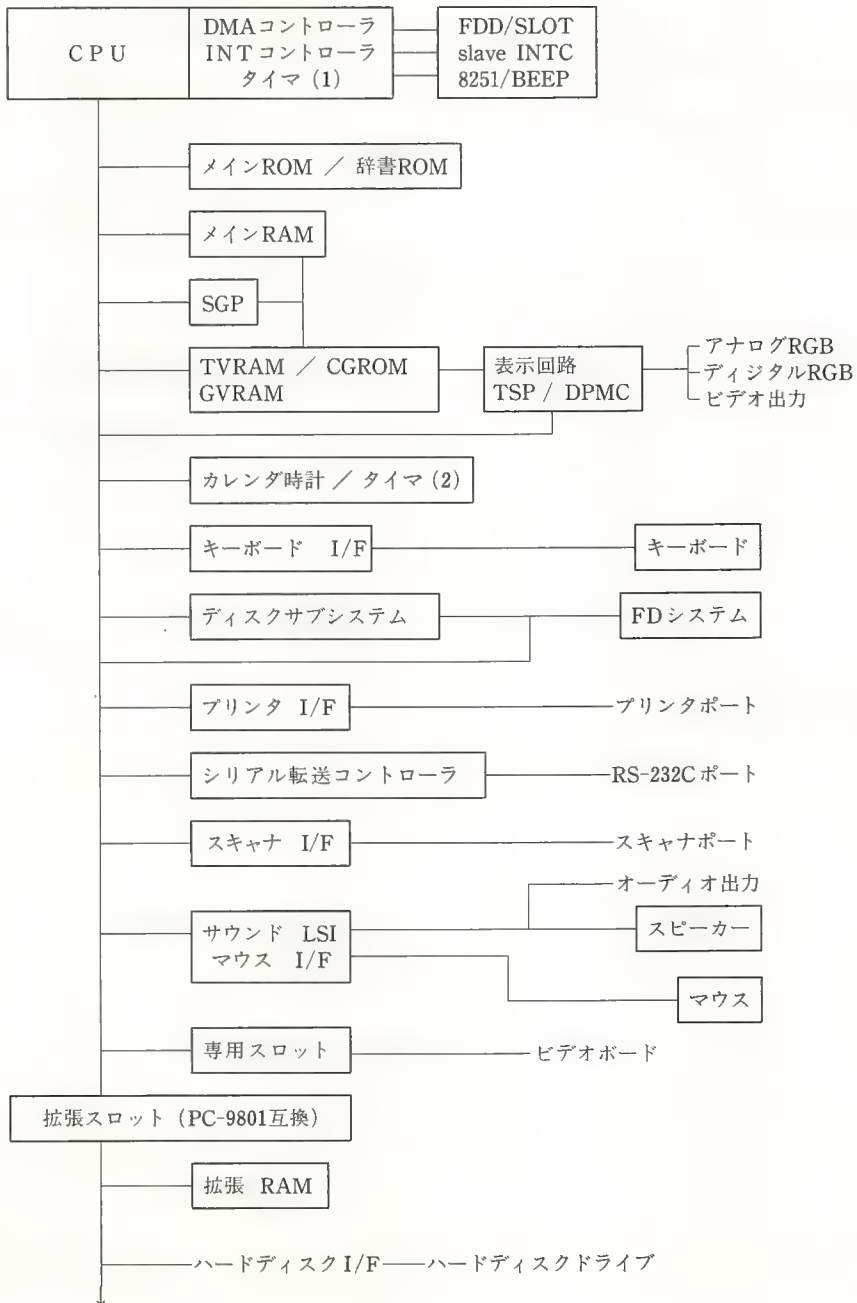
第8章 N₈₈ V1/V2モード 349

8.1 メモリ	349
8.2 テキスト表示	350
8.3 グラフィックス機能	352
8.4 画面合成機能	352

索引	354
----	-----

第1章 システム概説

1.1 ハードウェアブロック図



1.2 機能スペック

1) CPU

μPD9002

V30/μPD780 命令コンパチブル 8MHz

2) ROM

メイン ROM : 512KB(128KB 4バンク) E0000-FFFFFH

漢字 ROM : 288KB A0000-DFFFFH

ANK, 非漢字, 第1水準漢字, 第2水準漢字

辞書 ROM : 256KB A0000-DFFFFH

拡張 ROM 可能: 128KB 8バンク E0000-FFFFFH

3) RAM

標準 RAM : 512KB 00000-7FFFFH

拡張 RAM ボード(オプション): 80000-9FFFFH

メインメモリに割り当てられるのは最大 128KB

これを超えた部分はバンクメモリ(80000-9FFFFH)となる

TVRAM : 64KB A0000-AFFFFH

テキスト, スプライト表示用

GVRAM : 256KB A0000-DFFFFH

グラフィック表示用

CGRAM : 8KB

外字キャラクタジェネレータ用

JFP 学習機能用

メモリスイッチ用(64バイト)ライトプロテクト機能あり

バッテリーバックアップ

4) システム動作モード

V1/V2 モード 88F/M シリーズと同等のモード

V3 モード VA オリジナルのモード

5) キーボード

インターフェイス

インテリジェントタイプ

・マトリクススキャンインターフェイス 88M/F シリーズ互換

・キーコードインターフェイス 割込み使用

KB ユニット

インテリジェントシリアルキーボード(88MH/FH 互換)

オートリピートあり(キーコードインターフェイス時)

6) プリンタインターフェイス

セントロニクス社仕様標準 8ビットパラレルインターフェイス

1チャンネル

7) RS-232C インターフェイス

μPD8251 相当

RS-232C 規格準拠

1 チャンネル 調歩同期(非同期)式 300-9600 bps
外部同期可能

8) 割込み機能

V1/V2 モード時

8 レベル優先順位付き μ PD8241 相当

V3 モード時

13 レベル優先順位付き μ PD8259 相当

各チャンネル独立マスク可能

9) FDD

5 インチ 2HD/2D 兼用 2 ドライブ 外部増設不可 FDC: μ PD765

- ・インテリジェントインターフェイス

ディスクサブシステム使用

CPU μ PD780A 相当 クロック 4MHz

ROM 8KB

RAM 16KB

- ・DMA インターフェイス

DMA チャンネル 2 使用

10) DMA 機能

4 チャンネル 優先順位付き

チャンネル 0 バススロットに解放

チャンネル 1 未使用

チャンネル 2 FDD インターフェイス用

チャンネル 3 バススロットに解放

11) 表示機能

スプライト, テキスト, グラフィック 1, グラフィック 2 を優先順位付きで合成表示
透明色あり

画面マスク機能

外部ビデオ画面表示(オプション)

スプライト 8×4~256×256 ピクセル

1, 16 色/4096 色

最大 32 個 優先順位付き

テキスト 40×20~80×25

1, 16 色/4096 色

リバース, プリンク, シークレット, アンダーライン
文字種

ANK 8×8, 8×16 ドットを 256 字

漢 字 半角 8×16 漢字全角 16×16

非漢字, 第 1 水準, 第 2 水準

外 字 半角 8×16 全角 16×16

全角で 188 字

画面水平 4 分割可能(各分割画面にフレームバッファ設定)

水平垂直スムーズスクロール

グラフィックス 320×200～640×408
1, 16, 32 色/4096 色
256, 65536 色
独立 2 画面合成
画面水平 3 分割可能(各分割画面にフレームバッファ設定)
水平垂直スムーズスクロール(ラップラウンド付き)
カラーパレット 4096 色中 16/32 色指定. 各画面共通
プリンク機能あり

12) 描画機能

BitBlt 論理演算機能, 繰り返し Blt
LINE

13) ディスプレイ

15.98 KHz ノンインターレース 200/204 ライン
15.73 KHz インターレース 400/408 ライン
24.8 KHz ノンインターレース 400/408 ライン
デジタル RGB, アナログ RGB, ビデオ信号(NTSC)

14) 汎用タイマ

3 チャンネル モードにより 1・3 または 2・3 を使用可能
汎用タイマ 1 CPU 内蔵の TCU #0 : 周波数可変
汎用タイマ 2 88M/F シリーズ互換 : 600 Hz
汎用タイマ 3 マウスタイマ: 15, 30, 60, 120 Hz

15) カレンダー時計

μPD4990(88M/F シリーズ上位互換)
年/月/曜日/日/時/分/秒
大小の月, 閏年自動判別
バッテリーバックアップ

16) スキャナインターフェイス

μPD8255
8 ビットポート 3 組 双方向パラレルインターフェイス
1 チャンネル

17) サウンド

BEEP 1 声
PORT 1 声
SSG 音源 3 声
FM 音源 3 声
スピーカー/ボリューム/AUDIO OUT あり

18) スイッチ

システムモードスイッチ V1/V2 切換え
8 ビットディップスイッチ CRT モード等設定
スピードモードスイッチ 88M・F スピード/VA スピード切換え

19) マウスポート

88M/F シリーズコンパチブル

マウス/ジョイスティック/タブレット等接続用

入力4ビット 出力1ビット 入出力2ビット

20) 拡張スロット

PC-9801 互換2スロット/ビデオボード専用1スロット

電源容量 1スロット当り 0.5 A(5 V)

21) 固定ディスクユニットインターフェイス

オプション(98 シリーズ互換)

22) ビデオボード(オプション)

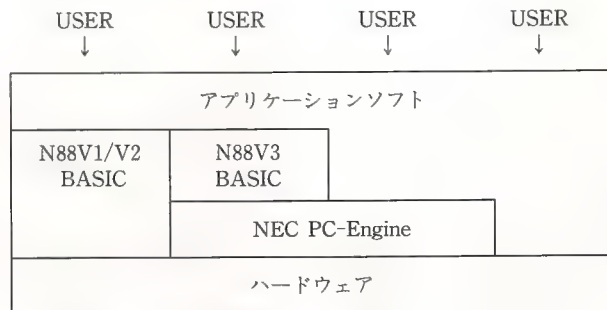
外部ビデオ信号のディジタル化機能

1.3 ソフトウェア動作環境

V3 モードでは NEC PC-Engine が稼動する。NEC PC-Engine は VA のハードウェア機能を有効に活用できる各種 BIOS(グラフィックス、テキスト、スプライト、サウンド、通信、周辺 I/O 等)や、日本語入力フロントプロセッサ、ファイル管理、プロセス管理、メモリ常駐型ソフトスーパーバイザ、日本語エディタなどをサポートするソフトウェアモジュールの集合体である。この中でもメモリ常駐型ソフトスーパーバイザは、メモリ常駐型アプリケーションソフトの作成が容易に実現できるような構造となっている。

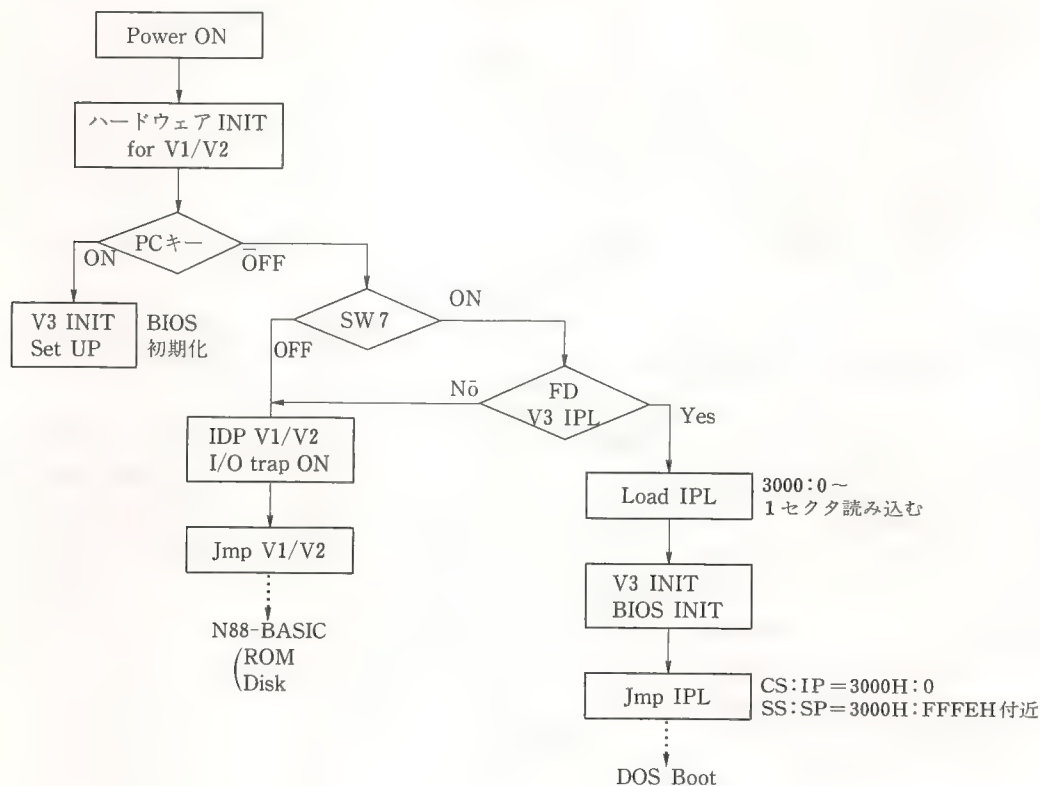
ファイルフォーマットは N88BASIC 形式と MS-DOS 形式をサポートする。テキストデータの漢字コードはシフト JIS を使用する。

VA におけるアプリケーションソフトの動作環境は下図のようになる。



1.4 システム起動プロセス

システムの起動時のフローは次のようになっている。



1.5 CPU

CPU(μPD9002)は V30 モードと μPD780 モードを持つオリジナル CPU である。両モードの関係は V30 のネイティブモードと 8080 エミュレーションモードとの関係と同じであり、相互のモード移行方法も同様となっている。

この CPU は、V30 とほぼ同じ命令セットを持っているが、INS・EXT・OUTM・INM の 4 命令(V30 ニーモニック)はサポートしていない。また、V50 と同様な DMA コントローラ・割込みコントローラ・タイマを CPU に内蔵している。

μPD780 モードでは μPD780 の全命令セットおよび未定義命令の多く、それに μPD780 モードから V30 モードへ移行するための RETEM・CALLN 命令をサポートしている。したがって、以前の 88 シリーズ用ソフトの大部分の動作が可能となっている。

〈ネイティブサービスルーチン〉

VA では μPD780 モード(V1/V2 モード)から V30 モードのメモリ・I/O をアクセスしたり、V30 (V3 モード)モードのプログラムに制御を移したりするためのサービスルーチンが用意されている。

① CALLN 91H(EDH, EDH, 91H)

V1/V2モードからV3モードのメモリ・I/Oアクセスを行う。

レジスタ設定：A=機能

bit 2 : 0 = メモリアクセス指定

1 = I/O アクセス指定

bit 1 : 0 = リード動作指定

1 = ライト動作指定

bit 0 : 0 = バイト動作指定

1 = ワード動作指定

HL = セグメント (メモリアクセス時のみ)

DE = オフセット (I/O アドレス)

BC = 値 (ライト・アウト時)

これらのレジスタを設定した後、CALLN 91Hを行うと、Aレジスタで指定した動作を行う。
リード・イン時にはBCレジスタに値が入る。

② CALLN 95H(EDH, EDH, 95H)

アドレス 1000H : E000H に用意されたユーザネイティブルーチンをCALLする。ここはV1/V2モードでのアドレスE000Hに相当するため、V1/V2モードでネイティブプログラムを用意してからこのファンクションを利用できる。ネイティブプログラムはIRET命令でμPD780モードに戻ることができる。

③ RETEM (EDH, FDH)

V30モードに戻る。V1/V2モードからV3モードに戻ると、V3モードとしての初期化後、1000H : C003H にジャンプする。

第2章 マップ

2.1 割込みベクタ

割込み番号	シンボル	機 能
0		CPU：0 除算割込み
1		CPU：シングルステップ
2		CPU：NMI
3		CPU：INT3。デバッグ用予約
4		CPU：INT 0。オーバーフロー割込み
5		CPU：境界エラー割込み
6		CPU：システム予約
7		CPU：システム予約
8		8259-1：汎用タイマ 1
9		8259-1：キーボード
A		8259-1：VRTC 割込み
B		8259-1：UINT 0 (Bus IR3)
C		8259-1：RS-232C
D		8259-1：UINT1 (Bus IR5)
E		8259-1：UINT2 (Bus IR6)
F		8259-1：――
10		8259-2：SGP (描画プロセッサ)
11		8259-2：ハードディスク (Bus IR9)
12		8259-2：UINT4 (Bus IR10)
13		8259-2：FDC (μ PD765)
14		8259-2：サウンド割込み
15		8259-2：マウスタイマ
16		8259-2：NDP 用予約
17		8259-2：予 約
18-1F		予 約
20		プロセス終了
21	\$Dos	システムコールサービス
22		プロセス終了アドレス
23		^C ブレーク処理
24		ディスクエラーハンドラ
25	\$ARead	アプソリュートディスクリード
26	\$AWrite	アプソリュートディスクライト
27	\$TSR	プロセスの常駐終了
28		予 約
29		ダイレクトコンソール出力用予約

割込み番号	シンボル	機 能
2A～32 33 34～3F	\$MBios	予 約 マウス BIOS 予 約
40～47		8214 : V3 モードでの 8214 割込みベクタ
40～7B		未使用
7C 7D 7E 7F		CPU : I/O Trap(IN) CPU : I/O Trap(OUT) CPU : LD A, R Trap 予 約
80 81 82 83 84 85 86 87 88 89 8A 8B 8C 8D 8E 8F	\$FDBios \$HDBios \$KYBios \$TXBios \$SRBios 予 約 予 約 \$AGBios \$ANBios \$PRBios \$CMBios \$SDBios \$TDBios \$JFBios \$LEBios \$SCBios	フロッピーディスク BIOS ハードディスク BIOS キーボード BIOS テキスト BIOS スプライト BIOS 予 約 予 約 拡張グラフィックス BIOS アニメーション BIOS プリンタ BIOS コミュニケーション BIOS サウンド BIOS カレンダー時計 BIOS 日本語フロントエンドプロセッサ BIOS ラインエディタ グラフィックス画面制御 BIOS
90 91 92 93 94 95 96 97 98 99 9A 9B 9C 9D 9E 9F	 \$Font \$PopUp \$ScnInl \$Auxbas \$Editor \$MonIf \$Shell	V1/V2 スーパーバイザ ネイティブモード メモリ・I/O アクセス ファンシーフォントサービス ポップアップサービス スクリーンエディタ (CALLN インターフェイス (V1/V2 スーパーバイザ用)) ポップアップキーボード BIOS インターフェイス V3 BASIC 内部使用 テキストエディタ シェル内部使用 V3 BASIC 内部使用 デバugg内部使用 日本語フロントエンドプロセッサ内部使用 BASIC-デバuggインターフェイス FCB ファイルネームの設定 内部コマンドの実行
A0 A1 A2 A3 A4 A5 A6	\$Dadd \$Dsub \$Dmul \$Ddiv \$Dcmp \$Dtol \$ItoD	浮動小数点加算 (BCD) 浮動小数点減算 (BCD) 浮動小数点乗算 (BCD) 浮動小数点除算 (BCD) 浮動小数点比較 (BCD) 浮動小数点→整数 整数→浮動小数点

割込み番号	シンボル	機 能
A7	\$Din	浮動小数点入力
A8	\$Dout	浮動小数点出力
A9-AD		予 約
AE	\$BasCo	内部使用
AF	\$IMath	マスパック初期化
B0	\$Dsin	浮動小数点 SIN (BCD)
B1	\$Dcos	浮動小数点 COS (BCD)
B2	\$Dtan	浮動小数点 TAN (BCD)
B3	\$Datn	浮動小数点 ATN (BCD)
B4	\$Dexp	浮動小数点 EXP (BCD)
B5	\$Dlog	浮動小数点 LOG (BCD)
B6	\$Dpwr	浮動小数点 POWER (BCD)
B7	\$Dsqr	浮動小数点 ROOT (BCD)
B8-BF		予 約
C0-FF		Free

2.2 I/O ポートマップ

●I/O ポートブロック

I/O アドレス	ブロック名
0000H } 00FFH	システムエリア 0 (88MH/FH 互換)
0100H } 01FFH	システムエリア 1
0200H } 02FFH	フレームバッファ制御エリア
0300H } 04FFH	カラーパレット制御エリア
0500H } 05FFH	GVRAM 制御エリア
0600H } 0FFFH	システムエリア 2 (リザーブ)
1000H } FEFFH	ユーザエリア
FF00H } FFFFH	システムエリア 3 (CPU 内部使用)

2.2.1 システムエリア 0

0000H 000EH	__KeyMap0 __KeyMapE	IN	B					
キーマトリクス								
	7	6	5	4	3	2	1	0
0000H	7	6	5	4	3	2	1	0
0001H	RETURN	.	,	=	+	*	9	8
0002H	G	F	E	D	C	B	A	@
0003H	O	N	M	L	K	J	I	H
0004H	W	V	U	T	S	R	Q	P
0005H	—	^]	¥	[Z	Y	X
0006H	7'	6&	5%	4\$	3#	2"	1!	0
0007H	_	/?	.>	,<	:+	:*	9)	8(
0008H	CTRL	SHIFT	カナ	GRPH	INSDEL	→	↑	CLR
0009H	ESC	SPACES	F5	F4	F3	F2	F1	STOP
000AH	CAPS	/	—	COPY	HELP	←	↓	TAB
000BH							R UP	R DOWN
000CH	DEL	INS	BS	F5	F4	F3	F2	F1
000DE	SPACE	決定	変換	F10	F9	F8	F7	F6
000EH			全角	PC	SFT RT	SFT LT	RET 10	RET FK

偶数番地からのワードアクセスも可能です。

0010H	__PrtData	OUT	B	
プリンタ／カレンダー時計インターフェイス				

●プリンタインターフェイス

	7	6	5	4	3	2	1 0
0010H	プリンタに出力するデータ						

●カレンダー時計

	7	6	5	4	3	2	1 0
0010H	—	—	—	—	CD0	C2	C1 C0

CD0 カレンダー時計データ出力

C2～C0 カレンダー時計モードコマンド

0 : レジスタホールド

1 : レジスタシフト

2 : タイムセット&カウンタホールド

3 : タイムリード

4 : 指定禁止

5 : 指定禁止

6 : テスト TP=2048Hz

7 : シリアルコマンドモード

参照：0040H, 01CBH CDI：データ入力

0020H	<u>ArtData</u>	I/O	B	
0021H	<u>ArtSts</u>	I/O	B	
RS-232C USART(μ PD8251)				

0020H データ

0021H モード/コマンド/ステータス

●モード設定(非同期モード)

	7	6	5	4	3	2	1	0
OUT 0021H	S2	S1	EP	PEN	L2	L1	B2	B1

S2	S1	ストップビット数	L2	L1	キャラクタ長
0	0	無効	0	0	5ビット
0	1	1ビット	0	1	6ビット
1	0	1.5ビット	1	0	7ビット
1	1	2ビット	1	1	8ビット
EP		パリティの奇遇	B2	B1	ボーレート
0		奇数	0	0	同期モード
1		偶数	0	1	×1
PEN		パリティイネーブル	1	0	×16
0		パリティなし	1	1	×64
1		パリティあり			

●コマンド設定(非同期モード)

	7	6	5	4	3	2	1	0
OUT 0021H	0	IR	RTS	ER	SBRK	RXE	DTR	TXEN

IR	内部リセット
0	ノーオペレーション
1	内部リセット
RTS	送信要求
0	RS-232 C コネクタの RTS 端子をローレベルにする
1	RS-232 C コネクタの RTS 端子をハイレベルにする
ER	エラーリセット
0	ノーオペレーション
1	エラーリセット
SBRK	ブレークキャラクタ送信
0	通常動作
1	RS-232 C コネクタの TxD 端子=1
RXE	受信許可
0	受信禁止
1	受信許可
DTR	データターミナルレディ
0	RS-232 C コネクタの DTR 端子をローレベルにする
1	RS-232 C コネクタの DTR 端子をハイレベルにする
TXEN	送信許可
0	送信禁止
1	送信許可

●ステータス

IN 0021H	7	6	5	4	3	2	1	0
	DSR	SYNDET	FE	OE	PE	TXE	RXRDY	TXRDY
DSR	データセットレディ							
0	RS-232 C コネクタの DSR 端子はローレベル							
1	RS-232 C コネクタの DSR 端子はハイレベル							
SYNDET	SYNC キャラクタ検出							
0	SYNC キャラクタ検出なし							
1	SYNC キャラクタ検出した							
FE	フレーミングエラー							
0	エラーなし							
1	フレーミングエラー発生							
OE	オーバーランエラー							
0	エラーなし							
1	オーバーランエラー発生							
PE	パリティエラー							
0	エラーなし							
1	パリティエラー発生							
TXE	送信バッファエンプティ							
0	フル							
1	エンプティ							
RXRDY	受信レディ							
0	ビジー							
1	レディ							
TXRDY	送信レディ							
0	ビジー							
1	レディ							

0030H	_MemSW1	IN	B	
メモリスイッチポート 1				

メモリスイッチの B1FC2H 番地をリードする特殊ポート

0030H	7	6	5	4	3	2	1	0
	MS17	MS16	MS15	MS14	MS13	MS12	MS11	MS10
MS17	未使用				MS13	テキスト表示行数		
	常に 1				0	25 行		
MS16	未使用				1	20 行		
	常に 1				MS12	テキスト表示文字数		
MS15	DEL コード処理				0	80 文字		
0	処理する				1	40 文字		
1	無視する				MS11	動作モード		
MS14	S パラメータ				0	ターミナルモード		
0	有効				1	BASIC		
1	無効				MS10	未使用		
						常に 1		

0030H	__Text8	OUT	B	
テキスト制御ポート 0				

	7	6	5	4	3	2	1	0
0030H	—	—	—	—	—	—	MCM	80CM

MCM モノクロモード
 0 カラー
 1 モノクロ
 80CM 表示文字数
 0 40 文字モード
 1 80 文字モード

0031H	__MemSW2	IN	B	
メモリスイッチポート 2				

メモリスイッチの B1FC6H 番地をリードする特殊ポート

	7	6	5	4	3	2	1	0
0031H	MS27	MS26	MS25	MS24	MS23	MS22	MS21	MS20

MS27	システムモード	MS23	ストップビット数
0	V2 モード	0	2 ビット
1	V1 モード	1	1 ビット
MS26	未使用	MS22	データビット長
	常に 1	0	8 ビット
MS25	通信モード	1	7 ビット
0	半二重	MS21	パリティ指定
1	全二重	0	偶数パリティ
MS24	X パラメータ	1	奇数パリティ
0	有効	MS20	パリティ有無
1	無効	0	パリティ有り
		1	パリティ無し

0031H		OUT	B	V1/V2 モード専用
システムポート 1				

	7	6	5	4	3	2	1	0
0031H	—	—	—	PM00	GDENO	RMODE	MMODE	VW1

PM00	グラフィックピクセルサイズ	MMODE	メモリモード
0	1 ビット/ピクセル	0	ROM/RAM モード
1	3 ビット/ピクセル	1	RAM モード
GDENO	グラフィック表示禁止	VW1	グラフィックス垂直解像度
0	禁止	0	400/408 ドット
1	許可	1	200/204 ドット
RMODE	ROM モード		
0	V1/V2 ROM		
1	モニタ ROM		

0032H	__GMode8	I/O	B	
システムポート 2				

	7	6	5	4	3	2	1	0
0032H	SINTM	GVAM	PMODE	TMODE	AVC2	AVC1	ROMSL1	ROMSL0

SINTM サウンドコントローラ割込みマスク
0 許可
1 禁止

GVAM GVRAM アクセスモード
0 独立アクセスモード
1 拡張アクセスモード
システムメモリモードが V3 モードのときには自動的に 0 となる。V3 モードの GVRAM アクセスモードの変更にはポート 0510H を用いる。

PMODE カラーモード
0 デジタルモード
1 アナログモード
ポート 010CH の PLTM 2 ビットが 0 のときだけ意味をもつ。

TMODE TVRAM 禁止
0 TVRAM 選択
1 TVRAM 禁止

AVC ビデオ出力制御
AVC2 AVC1
0 0 テレビ／ビデオモード
0 1 禁止
1 0 アナログ RGB モード
1 1 禁止
AVC はポート 0190H から設定可能。

ROMSL V1/V2 BASIC 拡張 ROM 選択
ROMSL1 ROMSL0
0 0 バンク 0
0 1 バンク 1
1 0 バンク 2
1 1 バンク 3

0034H		OUT	B	V1/V2 モード専用
GVRAM 制御ポート 1				

	7	6	5	4	3	2	1	0
0034H	—	ALU21	ALU11	ALU01	—	ALU20	ALU10	ALU00

ALU グラフィックス ALU モード
ALU01 ALU00
ALU11 ALU10
ALU21 ALU20
0 0 ビットリセット
0 1 ビットセット
1 0 ビット反転
1 1 NOP

0035H		OUT	B	V1/V2 モード専用
GVRAM 制御ポート 2				

	7	6	5	4	3	2	1	0
0035H	GAM	—	GDM1	GDM0	—	PLN2	PLN1	PLN0

GAM	GVRAM 選択		
0	メイン RAM 選択		
1	GVRAM 選択		
	システムメモリモードが V3 モードのときには自動的に 0 となる		
GDM	グラフィックスデータ制御		
	GDM1	GDM0	
	0	0	ALU 出力
	0	1	リードデータ
	1	0	プレーン 1→プレーン 0
	1	1	プレーン 0→プレーン 1
PLN2	プレーン 2 リード比較データ		
PLN1	プレーン 1 リード比較データ		
PLN0	プレーン 0 リード比較データ		

0040H	__TodCt1	OUT	B	
システムポート 3 (ストローブポート)				

	7	6	5	4	3	2	1	0
0040H	FBEEP	JOP1	BEEP	1	0	CCLK	CSTB	XPSTB

FBEEP	ポートサウンド出力							
JOP1	汎用出力ポート (正論理)							
BEEP	ブザー制御							
0	OFF							
1	ON							
CCLK	カレンダー時計クロックストローブ							
0	LOW							
1	HIGH							
CSTB	カレンダー時計ストローブ							
0	OFF							
1	ON							
XPSTB	プリンタストローブ							
0	ON							
1	OFF							

V3 モードでは BEEP ビットは常に 0 に設定しておき、01CDH の XBEEP で制御する。

0040H	__PrtSts	IN	B	
システムポート 4				

	7	6	5	4	3	2	1	0
0040H	1	1	VRTC	CD1	SW7	DCD	SW1	PBSY

VRTC CRT 垂直帰線区間

0 表示期間

1 垂直帰線期間

CDI カレンダ時計からの入力データ

SW7 インテリジェント FDD ブート制御(IFDSL)

0 インテリジェント FDD からブートする

1 インテリジェント FDD からブートしない

DCD RS-232C DCD

0 ON

1 OFF

SW1 CRT モード(CRTMD)

0 24.8 kHz

1 15 kHz

PBSY プリンタビジィ

0 レディ

1 ビジィ

SW1, DCD, CDI ビットはポート 01CBH から読める。

0044H	__YMSts1	I/O	B	
0045H	__YMDat1	I/O	B	
0046H	__YMSts2	I/O	B	
0047H	__YMDat2	I/O	B	
サウンドコントローラ(YM2203)				

	7	6	5	4	3	2	1	0
0044H	ステータス/レジスタアドレス							
0045H	リードデータ/ライトデータ							
	7	6	5	4	3	2	1	0
0046H	ステータス/レジスタアドレス(リザーブ)							
0047H	リードデータ/ライトデータ(リザーブ)							

● YM2203 ステータス

	7	6	5	4	3	2	1	0
IN 0044H	OPNBSY	—	—	—	—	—	TFLGB	TFLGA

OPNBSY

0 レディ

1 ビジィ

TFLGB タイマ B カウント終了フラグ

1 終了

TFLGA タイマ A カウント終了フラグ

1 終了

005CH		IN	B	V1/V2 モード専用
GVRAM ステータス				

	7	6	5	4	3	2	1	0
005CH	1	1	1	1	1	G2	G1	G0
G2	GVRAM プレーン2 選択ステータス							
0	禁止中							
1	選択中							
G1	GVRAM プレーン1 選択ステータス							
0	禁止中							
1	選択中							
G0	GVRAM プレーン0 選択ステータス							
0	禁止中							
1	選択中							

005CH		OUT	B	V1/V2 モード専用
005DH		OUT	B	
005EH		OUT	B	
005FH		OUT	B	
GVRAM 選択				

OUT 5CH GVRAM プレーン0 選択

OUT 5DH GVRAM プレーン1 選択

OUT 5EH GVRAM プレーン2 選択

OUT 5FH GVRAM アクセス禁止

* OUT されるデータは無関係

0070H		I/O	B	V1/V2 モード専用
メモリオフセット				

メモリウィンドウオフセットアドレスの設定/読出し

OUT 70H, オフセットアドレス

IN 70H

0071H		I/O	B	V1/V2 モード専用
拡張 ROM 選択				

	7	6	5	4	3	2	1	0
0071H	1	1	1	1	1	1	1	XEROM
XEROM	BASIC 拡張 ROM 選択							
1	選択							
0	禁止							

0078H		OUT	B	V1/V2 モード専用
メモリオフセットインクリメント				

メモリウィンドウのオフセットアドレスのインクリメント

OUT 78H (出力されるデータは無関係)

0080H	__HDDData	I/O	B	
0082H	__HDStS	I/O	B	
ハードディスク制御				

0080H ハードディスクデータ

0082H ハードディスクコマンド/ステータス

00BCH	__ScanA	I/O	B	
00BDH	__ScanB	I/O	B	
00BEH	__ScanC	I/O	B	
00BFH	__ScanD	I/O	B	
スキャナポート(μPD8255)				

00BCH データポート A

00BDH データポート B

00BEH データポート C

00BFH モードレジスタ

00E2H		I/O	B	V1/V2 モード専用
00E3H		I/O	B	
V1/V2 モード拡張 RAM 選択				

	7	6	5	4	3	2	1	0
00E2H	1	1	1	1	P1	P0	B1	B0

	7	6	5	4	3	2	1	0
00E3H	1	1	1	WE	1	1	1	RE

P1 P0 ページ選択
0 0 ページ 0
0 1 ページ 1
1 0 ページ 2
1 1 ページ 3
B1 B0 バンク選択
0 0 バンク 0
0 1 バンク 1
1 0 バンク 2
1 1 バンク 3

WE	拡張RAMバンクライト制御	
	OUT時	IN時
0	ライト禁止設定	ライト許可中
1	ライト許可設定	ライト禁止中
RE	拡張RAMバンクリード制御	
	OUT時	IN時
0	リード禁止設定	リード許可中
1	リード許可設定	リード禁止中

00E4H		OUT	B	V1/V2 モード専用
8214 モード割込み制御				

	7	6	5	4	3	2	1	0
00E4H	—	—	—	—	XSGS	XCST2	XCST1	XCST0
XCST2~0 INTC カレントステータス								
XSGS カレントステータスレジスタ制御								
0 有効								
1 無効								

00E6H		OUT	B	V1/V2 モード専用
8214 モード割込みマスク				

	7	6	5	4	3	2	1	0
00E6H	—	—	—	—	—	XRXRMF	XVRMF	XRTMF
XRXRMF RS-232C RXRDY 割込みマスク								
XVRMF VRTC 割込みマスク								
XRTMF 汎用タイマ2 割込みマスク								
0 禁止								
1 許可								

00E8H		I/O	B	(V1/V2 モード専用)
00E9H		I/O	B	
第一水準漢字フォント読出し				
+				

●アドレス出力(ワードアクセス可)

00E8H: CGROM 下位アドレス

00E9H: CGROM 上位アドレス

●パターンリード(ワードアクセス不可)

00E8H: パターン右側

00E9H: パターン左側

V3 モードではポート 14C~14FH を使用するか、CG をメモリ空間上にマッピングして使用する。

00ECH		I/O	B	V1/V2 モード専用
00EDH		I/O	B	
第二水準漢字フォント読出し				

●アドレス出力(ワードアクセス可)

00ECH: CGROM 下位アドレス

00EDH: CGROM 上位アドレス

●パターンリード(ワードアクセス不可)

00ECH: パターン右側

00EDH: パターン左側

V3 モードではポート 14C~14FH を使用するか、もしくは CG をメモリ空間上にマッピングして使用する。

00FCH	__IntFlpA	I/O	B	
00FDH	__IntFlpB	I/O	B	
00FEH	__IntFlpC	I/O	B	
00FFH	__IntFlpD	I/O	B	
インテリジェント FDD インターフェイス(μPD8255 サブセット)				

00FCH データ入力ポート

00FDH データ出力ポート

00FEH コントロールポート

00FFH モードレジスタ/コントロールポート操作

- データ入力ポート FD サブシステムからのデータを受け取る
- データ出力ポート FD サブシステムへデータを送出する
- コントロールポート(IN/OUT)

このポートの出力ビットのセット／リセットはポート FFH で行う。

	7	6	5	4	3	2	1	0
00FEH	ATNO	DACO	RFDO	DAVO	—	DACI	RFDI	DAVI

ATNO	ATN 出力
0	コマンドの出力ではない
1	コマンドを出力中
DACO	DAC 出力
0	ビジー
1	データ引取り完了
RFDO	RFD 出力
0	ビジー
1	ハンドシェイクレディ
DAVO	DAV 出力
0	データを出力していない
1	データを出力中
DACI	サブシステムからの DAC 入力
0	ビジー
1	サブシステムがデータ引取り完了
RFDI	サブシステムからの RFD 入力
0	ビジー
1	サブシステムハンドシェイクレディ
DAVI	サブシステムはデータを出力していない
1	サブシステムがデータを出力中

●コントロールポート操作ポート(OUT)

OUT FFH

① モードコマンド(初期設定): OUT 0FFH, 91H

② ビット操作コマンド

コントロールポート (FFH) の各出力ビットをビット単位に操作する。

ビット 7 常に 0

ビット 3～1 ビット位置設定

100 DAVO

101 RFDO

110 DACO

111 ATNO

ビット 0 セット／リセット設定

0 リセット

1 セット

2.2.2 システムエリア 1

0100H	__GrMode	I/O	W	
表示画面制御レジスタ				

	15	14	13	12	11	10	9	8
0101H	GDEN0	GVM	XVSP	SYNCEN	VMMD	DM	0	0
	7	6	5	4	3	2	1	0
0100H	RSM1	RSM0	GDEN1	SYNCM	0	0	VM1	VM0

GDEN0	グラフィック表示禁止レジスタ	
0	グラフィック表示禁止(リセット時)	
1	グラフィック表示イネーブル	
GVM	グラフィックスモードレジスタ	
0	グラフィック表示モード(リセット時)	
1	ビデオディジタルモード	
XVSP	ビデオ信号出力禁止レジスタ	
0	ビデオ信号出力禁止モード(リセット時)	
1	ビデオ信号出力モード(表示モード)	
SYNCEN	水平同期信号出力禁止レジスタ	
0	水平同期信号出力禁止モード(リセット時)	
1	水平同期信号出力モード(表示モード)	
DM	グラフィック表示モードレジスタ(153H GMSP 参照)	
0	マルチプレーンモード(リセット時)	
1	シングルプレーンモード	
VMMD	GVRAM モード(シングルプレーン時)	
0	1 画面モード(グラフィック画面 0 のみ)(リセット時)	
1	2 画面モード(グラフィック画面 0/1)	
RSM	CRT のラスタスキャンモード	
RSM1	RAM0	
0	0	ノンインターレースモード 0 (リセット時)
0	1	ノンインターレースモード 1
1	0	インターレースモード 0
1	1	インターレースモード 1
GDEN1	グラフィック表示回路リセット	
0	グラフィック表示回路リセット(リセット時)	
	(表示関係のポートの設定は保証されない)	
1	グラフィック表示モード	
SYNCM	同期信号モード	
0	内部同期モード(リセット時)	
1	外部同期モード	
VM	垂直解像度	
VW1	VW1	
0	0	400 ドット(リセット時)
0	1	408 ドット
1	0	200 ドット
1	1	204 ドット

0102H	__GrRes	I/O	W	
グラフィック画面制御レジスタ				

	15	14	13	12	11	10	9	8
0103H	0	0	0	HW1	0	0	PM11	PM10
	7	6	5	4	3	2	1	0
0102H	0	0	0	HW0	0	0	PM01	PM00

HW1	グラフィック画面 1 水平解像度	
0	640 ドット (リセット時)	
1	320 ドット	
PM1	グラフィック画面 1 ピクセルサイズ	
PM11	PM10	
0	0	1 ビット/ピクセル (リセット時)
0	1	4 ビット/ピクセル
1	0	8 ビット/ピクセル
1	1	指定禁止
HW0	グラフィック画面 0 水平解像度	
0	640 ドット (リセット時)	
1	320 ドット	
PM0	グラフィック画面 0 ピクセルサイズ	
PM01	PM00	
0	0	1 ビット/ピクセル (リセット時)
0	1	4 ビット/ピクセル
1	0	8 ビット/ピクセル
1	1	16 ビット/ピクセル

0106H	__ColComp	OUT	W	
パレット指定画面制御レジスタ				

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
優先表示画面 3				優先表示画面 2				優先表示画面 1				優先表示画面 0			
〈各ニブルの値〉				〈指定画像ソース〉											
0				指定なし(表示 OFF)											
8				テキスト画面											
9				スプライト画面											
A				グラフィック画面 0											
B				グラフィック画面 1											

0108H	__RGBComp	OUT	W	
直接色指定画面制御レジスタ				

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	優先表示画面 5				優先表示画面 4			
〈各ニブルの値〉				〈指定画像ソース〉											
0				指定なし(表示 OFF)											
8				グラフィック画面 0											
9				グラフィック画面 1											

010AH	__MskMode	OUT	W	
画面マスクモードレジスタ				

	15	14	13	12	11	10	9	8
010BH	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
010AH	0	0	GMP1	GMP0	MKM11	MKM10	MKM01	MKM00

(リセット時不定)

GMP マスク挿入位置

GMP1	GMP0	
0	0	表示画面 0 と表示画面 1 との間
0	1	表示画面 1 と表示画面 2 との間
1	0	表示画面 2 と表示画面 3 との間
1	1	表示画面 3 と表示画面 4 との間

MKM1 マスクモード (外側)

MKM11	MKM10	
0	0	ノーオペレーション
0	1	直後の低優先画面を透明にする
1	0	高優先画面をすべて透明にする
1	1	低優先画面をすべて透明にする

MKM0 マスクモード (内側)

MKM01	MKM00	
0	0	ノーオペレーション
0	1	直後の低優先画面を透明にする
1	0	高優先画面をすべて透明にする
1	1	低優先画面をすべて透明にする

010CH	__PalMode	OUT	W	
カラーパレットモードレジスタ				

	15	14	13	12	11	10	9	8
010DH	0	0	BDM1	BDM0	0	0	0	PLTM2
	7	6	5	4	3	2	1	0
010CH	PLTM1	PLTM0	PLTP1	PLTP0	BLKM1	BLKM0	BLKD1	BLKD0

(リセット時不定)

BDM バックドロップモード

BDM1	BDM0	
0	0	バックドロップモード 0 (内バックドロップカラー 外透明)
0	1	バックドロップモード 1 (内透明 外バックドロップカラー)
1	0	バックドロップモード 2 (内外 バックドロップカラー)
1	1	バックドロップモード 3 (内外 透明)

※ 24.8KHz CRT 使用時はモード 0 のみ可能

PLTM カラーパレットモード

PLTM2	PMODE	PM00	(PMODE は 32H, PM00 は 31H)
0	0	0	カラーパレットモード 1
0	0	1	カラーパレットモード 2
0	1	0	カラーパレットモード 0
0	1	1	カラーパレットモード 2

PLTM2	PLTM1	PLTM0	
1	0	0	カラーパレットモード 0 (パレットセット 0 使用)
1	0	1	カラーパレットモード 1 (パレットセット 1 使用)
1	1	0	カラーパレットモード 2 (混在モード)
1	1	1	カラーパレットモード 3 (同時 32 色モード)
PLTP	混在モードでパレットセット 1 を使う表示画面の指定		
PLTP1	PLTP0		
0	0		テキスト画面
0	1		スプライト画面
1	0		グラフィック画面 0
1	1		グラフィック画面 1
BLKM	カラーブリンクモード		
BLKM1	BLKM0		
0	0		ブリンク OFF
0	1		ブリンク周期 32 フレーム
1	0		ブリンク周期 64 フレーム
1	1		ブリンク周期 128 フレーム
BLKD	ブリンクデューティ		
BLKD1	BLKD0		
0	0		12.5%
0	1		25%
1	0		50%
1	1		75%

010EH	__DropCol	OUT	W	
バックドロップカラーレジスタ				

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
010EH	G3	G2	G1	G0	*	*	R3	R2	R1	R0	*	B3	B2	B1	B0	*
	G 領域						R 領域					B 領域				

[*]は RGB 各領域の上位ビットがすべて 0 のとき 0 に、それ以外のときは 1 に設定する。

0110H	__PageMsk	OUT	W	
カラーコード／プレーンマスクレジスタ				

	15	14	13	12	11	10	9	8
0111H	TSCR3	TSCR2	TSCR1	TSCR0	GCF3	GCF2	GCF1	GCF0
	7	6	5	4	3	2	1	0
0110H	G3MSK	88MD	0	0	G3SW	G2SW	G1SW	G0SW

(リセット時)

TSCR	テキスト／スプライト判別境界カラー カラーコードが 1～設定値 : スプライト カラーコードが 1～設定値+1～15 : テキスト							
GCF	グラフィックスフォアグラウンドカラーレジスタ シングルプレーン, 1 ビット／ピクセル時のカラー							
G3MSK	マルチプレーン 4 ビット／ピクセル プレーン 3 画面スイッチ							
0	OFF							
1	ON							

88MD	グラフィック表示回路モードの指定		
0	V3 モード		
1	V1/V2 モード		
GnSW	マルチプレーン	1ビット/ピクセル	プレーン n 画面スイッチ
0	OFF		
1	ON		

0124H	__XParG0	OUT	W	
グラフィック画面 0 透明色レジスタ				

	15	14	13	12	11	10	9	8
0125H	GOTC15	GOTC14	GOTC13	GOTC12	GOTC11	GOTC10	GOTC9	GOTC8
	7	6	5	4	3	2	1	0
0124H	GOTC7	GOTC6	GOTC5	GOTC4	GOTC3	GOTC2	GOTC1	GOTC0

GOTCn	グラフィック画面 0	カラーパレット n の透明色指定
0	不透明色指定	
1	透明色指定	

0126H	__XParG1	OUT	W	
グラフィック画面 1 透明色レジスタ				

	15	14	13	12	11	10	9	8
0127H	G1TC15	G1TC14	G1TC13	G1TC12	G1TC11	G1TC10	G1TC9	G1TC8
	7	6	5	4	3	2	1	0
0126H	G1TC7	G1TC6	G1TC5	G1TC4	G1TC3	G1TC2	G1TC1	G1TC0

G1TCn	グラフィック画面 1	カラーパレット n の透明色指定
0	不透明色指定	
1	透明色指定	

012EH	__XParTxt	OUT	W	
テキスト/スプライト透明色レジスタ				

	15	14	13	12	11	10	9	8
012FH	TSTC15	TSTC14	TSTC13	TSTC12	TSTC11	TSTC10	TSTC9	TSTH8
	7	6	5	4	3	2	1	0
012EH	TSTC7	TSTC6	TSTC5	TSTC4	TSTC3	TSTC2	TSTC1	1

TSTCn	テキスト/スプライト	カラーパレット n の透明色指定
0	不透明色指定	
1	透明色指定	

0130H	__MskLeft	OUT	W	
0132H	__MskRit	OUT	W	
0134H	__MskTop	OUT	W	
0136H	__MskBot	OUT	W	
画面マスクパラメータ				(リセット時不定)

●画面マスク左右：640ドット解像度の画面に対する値

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0130H	0	0	0	0	0	0	マスク 左端									
0132H	0	0	0	0	0	0	マスク 右端									

●画面マスク上下：200／204ドット解像度の画面に対する値

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0134H	0	0	0	0	0	0	0	0	マスク 上端							
0136H	0	0	0	0	0	0	0	0	マスク 下端							

0142H		I/O	B	
0146H		I/O	B	
テキストコントローラ (TSP)				

●コマンドポート (OUT)

	7	6	5	4	3	2	1	0
0142H	コ マ ン ド							

●ステータスポート (IN)

	7	6	5	4	3	2	1	0
0142H	LP	VB	SC	ER	EMEN	BUSY	OBF	IBF

- LP ライトペン信号検出(VA では使用不可)
- VB 垂直消去期間
- SC スプライトコントロール(スプライトオーバー／衝突)
- ER エラー発生
- EMEN エミュレーション展開実行中
- BUSY コマンド実行中
- OBF 出力データバッファフル
- IBF 入力データバッファフル(コマンド／パラメータ共通)

0148H	__TxtMode	OUT	B	
テキスト制御ポート				

	7	6	5	4	3	2	1	0
0148H	TD	VALT2	VALT1	VALT0	ATM	ANKM	TVWM	1
TD	テキスト表示禁止機能							
0	表示 ON							
1	表示 OFF							
VALT	TVRAM アクセス制限パラメータ(リセット時=000)							
	設定値=0 制限なし							
	0 以外 設定値×4 回まで							
ATM	テキストアトリビュートモード							
0	V3 モード(リセット時)							
1	V1/V2 モード							
ANKM	ANK 文字フォントモード							
0	16 ドットモード(リセット時)							
1	8 ドットモード							
TVWM	TSP メモリモード							
0	バイトモード							
1	ワードモード(リセット時)							

014CH	__CGAddr	OUT	W	
014EH	__CGData	I/O	B	
014FH	__CGRow	OUT	B	
漢字 CG ポート				

●文字指定

OUT 14CH (ワード)：ハードウェア文字コード

- ・2バイト文字(日本語文字)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	JIS 第二バイト								0	JIS 第一バイト-20H					

- ・1バイト文字(ANK 文字)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	1バイトコード						

OUT 14FH (バイト)：ラスト番号／フォント左右

	7	6	5	4	3	2	1	0
014FH	—	—		—	ラスト番号			
				└─文字の左のとき 1				
				└─文字の右のとき 0				

●フォントアクセス

IN 14EH (バイト)：フォントデータ

(外字領域の場合書込みも可能)

0150H	__V1V2	IN	W	
システム動作モード				

	15	14	13	12	11	10	9	8
0151H	1	1	1	1	1	1	1	1
	7	6	5	4	3	2	1	0
0150H	1	1	1	1	1	1	X88V2	X88V1

X88V2	X88V1	システム動作モード
0	0	予約
1	0	V1 モード
0	1	V2 モード
1	1	予約

V3 モードはソフトウェア的に移行する。

0152H	__MemMode	I/O	W	
メモリマップレジスタ				

	15	14	13	12	11	10	9	8
0153H	0	SMM	0	GMSP	SMBC3	SMBC2	SMBC1	SMBC0
	7	6	5	4	3	2	1	0
0152H	RBC13	RBC12	RBC11	RBC10	RBC03	RBC02	RBC01	RBC00

SMM	システムメモリモード	
0	V1/V2 モード	
1	V3 モード(リセット時)	
GMSP	グラフィック VRAM 描画モード (101H DM 参照)	
0	マルチプレーンモード(リセット時)	
1	シングルプレーンモード	
SMM が V1/V2 モードの場合はこのビットを 1 にしてはならない。		
SMBC3~0	システムメモリエリアバンクコントロール	
0	バススロット	
1	TVRAM (リセット時)	
2	リザーブ	
3	リザーブ	
4	GVRAM	
5	リザーブ	
6	リザーブ	
7	リザーブ	
8	漢字 ROM #1	
9	漢字 ROM #2	
A	リザーブ	
B	リザーブ	
C	辞書 ROM	
D	リザーブ	
E	リザーブ	
F	リザーブ	
RBC13~10	ROM1 バンクコントロール	
0	ROM10 選択 (リセット時)	
1	ROM11 選択	
2~5	禁止	
6	リザーブ	
7	リザーブ	
8~F	バススロットに解放	
RBC03~00	ROM0 バンクコントロール	
0	ROM00 選択 (リセット時)	
1	ROM01 選択	
2	ROM02 選択	
3	ROM03 選択	
4	ROM04 選択	
5	ROM05 選択	
6	リザーブ	
7	リザーブ	
8~F	バススロットに解放	

0154H	__RefMode	OUT	B	
リフレッシュモードレジスタ				

	7	6	5	4	3	2	1	0
0154H	0	0	0	0	0	0	RFSHM1	RFSHM0

RFSHM1	RFSHM0	リフレッシュモード
0	0	1 サイクルに 1 アドレス (リセット時)
0	1	1 サイクルに 4 アドレス
1	0	1 サイクルに 8 アドレス (通常)
1	1	1 サイクルに 16 アドレス

このポートはシステムイニシャライズ時に設定され、以後変更してはならない。

0156H		OUT	B	
ROM バンクステータス				

	7	6	5	4	3	2	1	0
0156H	RBS7	RBS6	RBS5	RBS4	RBS3	RBS2	RBS1	RBS0

RBSn	0=バンク n に ROM が実装されていて、 オートスタートを要求している。 1=ROM はない/オートスタートの必要がない。
------	--

ROM バンク 18~1FH (アドレス F0000~FFFFFFH) の 8 バンクのオートスタート要求ビット。スロットに拡張された ROM がシステム起動時にオートスタートを要求する場合には、このポートへのリードアクセスに対して該当するビットを 0 とする。

0158H	__IntMode	OUT	B	
割り込みモード変更				

OUT 158H データは無関係

割り込みモードを 8214 モードから 8259 モードに変更する。

一度 8259 モードに変更すると 8214 モードに戻すことはできない。

015CH	__NoNMI	OUT	B	
015EH	__OkNMI	OUT	B	
NMI マスクポート (ストローブポート)				

OUT 015CH : NMI 禁止 (通常)

OUT 015EH : NMI 許可

通常 NMI は禁止にしておく。

0160H	__DmaRes	OUT	B	
0161H	__ChanSel	I/O	B	
0162H	__DmaCnt	I/O	W	
0164H	__DmaLo	I/O	W	
0166H	__DmaHi	I/O	B	
0168H	__DmaCtl	I/O	W	
016AH	__DmaMode	I/O	B	
016BH	__DmaSts	IN	B	
016FH	__DmaMask	I/O	B	
DMA コントローラ				

CH0 リザーブ(拡張スロット)

CH1 未使用

CH2 FDD データ

CH3 リザーブ(拡張スロット)

●リセットコマンド (DMAC)

	7	6	5	4	3	2	1	0
OUT 0160H	—	—	—	—	—	—	—	RES

RES 内部レジスタのリセット

0 NOP

1 リセット

●チャンネルレジスタ (DMAC)

	7	6	5	4	3	2	1	0
OUT 0161H	—	—	—	—	—	BASE	SELCH1	SELCH0

BASE ベースオンリー

0 リード時 カレント選択

ライト時 カレント/ベース選択

1 リード/ライト時 ベース選択

SELCH1 SELCH0 チャンネル選択

0 0 チャンネル0

0 1 チャンネル1

1 0 チャンネル2

1 1 チャンネル3

	7	6	5	4	3	2	1	0
IN 0161H	—	—	—	BASE	SEL3	SEL2	SEL1	SEL0

BASE 出力時と同じ

SEL3~0 選択されているチャンネル

0 0 0 1 チャンネル0

0 0 1 0 チャンネル1

0 1 0 0 チャンネル2

1 0 0 0 チャンネル3

●カウントレジスタ (DMAC)

I/O	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0162H	C15 - C0															

●アドレスレジスタ (DMAC)

I/O	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0164H	A15 - A0															
0166H									0	0	0	0	A19-A16			

●デバイスコントロールレジスタ (DMAC)

I/O	15	14	13	12	11	10	9	8
0169H	—	—	—	—	—	—	WEV	BHLD
I/O	7	6	5	4	3	2	1	0
0168H	—	—	0	ROT	—	DDMA	—	—

WEV ペリファイ時のウェイトの禁止

0 ウェイト挿入禁止

1 ウェイト挿入許可

ROT 回転優先指定

0 固定優先

1 回転優先

BHLD バスモード

0 バスリリースモード

1 バスホールドモード

DDMA DMA 動作の禁止

0 動作許可

1 動作禁止

●モードコントロールレジスタ (DMAC)

I/O	7	6	5	4	3	2	1	0
016AH	TMOD1	TMOD0	ADIR	AUT1	TDIR1	TDIR0	—	W/XB

TMOD1, 0

0 0

転送モード

デマンドモード

0 1

シングルモード

1 0

ブロックモード

1 1

指定禁止(カスケードモード)

TDIR1, 0

0 0

転送方向

ペリファイモード

0 1

I/O → メモリ

1 0

メモリ → I/O

1 1

指定禁止

ADIR

0

アドレス更新方向

インクリメント

1

デクリメント

W/XB

0

ワードバイトモード

バイトモード

1

指定禁止(ワードモード)

AUT1

0

オートイニシャライズ

オートイニシャライズ禁止

1

オートイニシャライズ許可

●ステータスレジスタ (DMAC)

IN	7	6	5	4	3	2	1	0
016BH	RQ3	RQ2	RQ1	RQ0	TC3	TH2	TC1	TC0

RQ3~0

0

DMA リクエストステータス(チャンネル 3~0)

リクエスト無し

1

リクエスト有り

TC3~0

0

ターミナルカウントステータス(チャンネル 3~0)

未終了

1

END またはターミナルカウント

●マスキレジスタ (DMAC)

I/O	7	6	5	4	3	2	1	0
016FH	—	—	—	—	M3	M2	M1	M0

M3～0 DMA リクエストマスク (チャンネル 3～0)

0 DMA 要求許可

1 DMA 要求禁止

0184H	__SlaICU0	I/O	B	
0186H	__SlaICU1	I/O	B	
0188H	__MasICU0	I/O	B	
018AH	__MasICU1	I/O	B	
割込みコントローラ				

186H, 18AH マスタ ICU

184H, 186H スレーブ ICU

参照: 158H, 1CDH, 1CFH

●割込みマスク

18AH マスタ ICU 割込みマスク

186H スレーブ ICU 割込みマスク

IN/OUT	7	6	5	4	3	2	1	0
018AH 0186H	M7	M6	M5	M4	M3	M2	M1	M0

Mn: 割込みマスク 各レベルの割込みを禁止/許可する。

0 割込み許可

1 割込み禁止

0190H	__VDMoDe	I/O	B	
システムポート 5				

	7	6	5	4	3	2	1	0
0190H	0	0	0	FBEN	AVC2	AVC1	0	RSTMD

RSTMD リセットステータス

0 パワーオンリセット時

1 マニュアルリセット時

(パワーオンリセット後, ソフトウェアによりセットすること)

AVC2 AVC1 ビデオ出力制御

0 0 テレビ/ビデオ モード (オフライン)

0 1 指定禁止

1 0 アナログ RGB モード (リセット時)

1 1 指定禁止

FBEN FBEEP イネーブル

0 FBEEP 禁止

1 FBEEP 許可 (リセット時)

0197H	__KeyMode	OUT	B	
キーボードサブ CPU コマンドポート				

	15	14	13	12	11	10	9	8
0197 H	1	0	0	0	0	FCLR	RTRY	RESET
0197H	1	1	0	KTARY	KCIFE	AREP	PRIK	FCTRL
FCLR	FIFO クリア				KCIFE	キーコードインタフェース動作モード		
0	ノーオペレーション				0	使用しない (リセット時)		
1	FIFO クリア				1	使用する		
RTRY	再送要求				AREP	オートリピート動作		
0	ノーオペレーション				0	指定しない (リセット時)		
1	再送要求				1	指定する		
RESET	サブ CPU リセット				PRIK	優先キー先行送信		
0	ノーオペレーション				0	指定しない (リセット時)		
1	リセット				1	指定する		
KTARY	キー配列指定				FCTRL	FIFO 制御		
0	JIS 配列 (リセット時)				0	FIFO 使用せず (リセット時)		
1	50 音配列				1	FIFO 使用		

0198H	__MemPro	OUT	B	
019AH	__MemWrt	OUT	B	
メモリスイッチ書込み禁止/許可				

● ストロープポート

OUT 0198H メモリスイッチ書込み禁止 (リセット時)

OUT 019AH メモリスイッチ書込み許可

01A0H	__Timer0	I/O	B	
01A2H	__Timer1	I/O	B	
01A4H	__Timer2	I/O	B	
01A6H	__TimMode	I/O	B	
TCU (タイマカウンタユニット)				

● TCU カウンタポート

	7	6	5	4	3	2	1	0
01A0H I/O	カウンタ#0 カウント値/ステータス							
01A2H I/O	カウンタ#1 カウント値/ステータス							
01A4H I/O	カウンタ#2 カウント値/ステータス							

カウンタ0 汎用タイマ1

カウンタ1 BEEP 周波数設定

カウンタ2 RS-232C ボーレート設定

<ステータス>

	7	6	5	4	3	2	1	0
IN	OL	NC	RWM1	RWM0	CMODE2	CMODE1	CMODE0	BDM
	OL カウンタの出力信号レベル				NC カウントデータ無効フラグ			
	0	ローレベル			0	カウントデータ有効(カウント中)		
	1	ハイレベル			1	カウントデータ無効		
					0	設定されているモードがリードできる		

● TCU 制御ポート (01A6H)

このポートで指定後、カウンタの設定・カウンタ値/ステータスの読出しを行う。

① カウンタの設定

	7	6	5	4	3	2	1	0
OUT	SC1	SC0	RWM1	RWM0	CMODE2	CMODE1	CMODE0	BDM
SC1	SC0	設定対象カウンタ設定						
0	0	カウンタ#0						
0	1	カウンタ#1						
1	0	カウンタ#2						
1	1	(マルチプルラッチコマンド③参照)				CMODE	カウンタモード	
					000	モード0(カウンタ#0)		
					×11	モード3(カウンタ#0, 1, 2)		
RWM1	RWM0	リードライトモード						
0	0	(カウントラッチコマンド②参照)						
0	1	下位1バイト				DB	カウント動作指定	
1	0	上位1バイト				0	バイナリカウント	
1	1	下位, 上位の順の2バイト				1	BCD カウント	

② カウントラッチ (1つのカウンタのリード)

	7	6	5	4	3	2	1	0
OUT	SC1	SC0	0	0	0	0	0	0
	SC1	SC0	リード対象カウンタの選択					
	0	0	カウンタ#0					
	0	1	カウンタ#1					
	1	0	カウンタ#2					

このコマンドで指定してポート 1A0H~1A4H でリードする。

③ マルチプルラッチコマンド (複数カウンタのリード)

	7	6	5	4	3	2	1	0
OUT	1	1	XCL	XSL	CT2	CT1	CT0	0
	XCL	カウンタラッチモード						
	0	ラッチする						
	1	ラッチしない						
	XSL	ステータス ラッチモード						
	0	ラッチする						
	1	ラッチしない						
	CT2	カウンタ#2の選択						
	CT1	カウンタ#1の選択						
	CT0	カウンタ#0の選択						
	0	選択しない						
	1	選択する						

01A8H	__TimFreq	OUT	B	
汎用タイマ3 制御ポート				

	7	6	5	4	3	2	1	0
01A8H	MINTEN	0	0	0	0	0	MTP1	MTP0

MINTEN 汎用タイマ3 割込みイネーブル

0 禁止 (リセット時)

1 許可

タイミングによってはこのビットで割込み禁止にした後で、1回だけ割込みが発生する
場合がある。

MTP1 MTP0 汎用タイマ3 インターバル

0 0 120Hz (リセット時)

0 1 60Hz

1 0 30Hz

1 1 15Hz

01B0H	__DskMode	OUT	B	
FDC モードレジスタ				

	7	6	5	4	3	2	1	0
01B0H	0	0	0	0	0	0	0	MODE

MODE FDD インターフェイス 動作モード

0 インテリジェントモード (リセット時)

1 DMA モード

01B2H	__DskCt1	OUT	B	
FDC 制御ポート0 (DMA モード時のみアクセス可)				

	7	6	5	4	3	2	1	0
01B2H	0	0	CLK	DSI	TD1	TD0	RV1	RV0

CLK FDC クロック選択

0 4.8MHz (リセット時)

1 8MHz

DSI FDC のドライブセレクトの禁止

0 許可 (リセット時)

1 禁止

TD1/ TD0 ドライブ1/0 トラック密度

0 48 TPI (リセット時)

1 96 TPI

RV1/ RV0 ドライブ1/0 モードセレクト

0 2D, 2DD モード (リセット時)

1 2HD モード

(注) ドライブのトラック密度を 48tpi から 96tpi に変更する場合には、まず DSI=1 にした後
切換えを行う必要がある。

01B4H	__MtrCt1	OUT	B	
FDC 制御ポート 1 (DMA モード時のみアクセス可)				

●モータコントロール

	7	6	5	4	3	2	1	0
01B4H	0	0	0	0	PCM	0	M1	M0

- PCM プリシフト指定
- 0 プリシフトを行わない(リセット時)
- 1 プリシフトを行う
- M1/M0 ドライブ 1/0 モータ制御
- 0 モータ OFF(リセット時)
- 1 モータ ON
- (注) モータ ON の際は、動作可能となるまでに 600ms の待ち時間を必要とする。

01B6H	__DskMisc	I/O	B	
FDC 制御ポート 2 (DMA モード時のみアクセス可)				

●FD コントロール

	7	6	5	4	3	2	1	0
OUT 01B6H	FDCRST	FDCFRY	FRYCEN	DMAE	0	XTMASK	0	TTRG

- FDCRST FDC リセット
- 0 ノーオペレーション
- 1 リセット
- FDCFRY FRYCEN FDC 強制レディ制御
- X 以前の状態を保持
- 0 強制レディ状態解除
- 1 強制レディ状態設定
- DMAE DMA 制御
- 0 DMA 禁止
- 1 DMA 許可
- XTMASK FDC タイマ割込みマスク
- 0 割込み禁止
- 1 割込み許可
- TTRG FDC タイマトリガ
- 0 FDC タイマクリア
- 1 FDC マイマスタート

●ステータス

	7	6	5	4	3	2	1	0
IN 01B6H	—	—	—	RDY	—	—	—	—

- RDY FDD レディ
- 0 ビジィ
- 1 レディ

01B8H	__FDCSts	IN	B	
01BAH	__FDCData	I/O	B	
FDC(μPD765) ポート				

01B8H FDC ステータスポート
01BAH FDC データポート

01C1H	__KeyData	IN	B	
キーボードデータポート(キーコードモード)				

	7	6	5	4	3	2	1	0
01C1H	キーコード データ							

01C9H	__Switch	IN	B	
01CBH	__MiscInp	IN	B	
01CDH	__IntMask	I/O	B	
01CFH	__SysPort	OUT	B	???
システムポート 6, 7, 8 (μPD8255)				

●システムポート 6 (IN)

	7	6	5	4	3	2	1	0
01C9H	1	1	SPEED	SW5	SW4	SW3	SW2	1

SPEED スピードモード
0 Sモード
1 Hモード
SW5～2 ディップスイッチ 5～2 の値
0 オン
1 オフ

●システムポート 7 (IN)

	7	6	5	4	3	2	1	0
01CBH	CI	CTS	DCD	UINT3	XSW1	0	0	CDI

CI RS-232C 被呼信号
0 ON
1 OFF
CTS RS-232C 送信可能信号
0 ON

第3章 メモリ

3.1 メモリモード

本機には CPU が直接アクセスできる 1M バイトのメモリ空間があるが、そのメモリ割付けには V3 モードと V1/V2 モードの 2 種類がある。

- (1) V3 モード : VA オリジナルモード
- (2) V1/V2 モード : 88 コンパチブルモード

本章では主に V3 モードについて記述する。

3.2 メモリレイアウト

V3 モードでは 1M バイトのメモリ空間すべてを使用し、その大まかなレイアウトは下図のようになっている。このうち、ROM エリア・システムメモリエリア・RAM エリアの一部はバンク構成になっている。

● ROM エリア

FFFFFH	ROM10	ROM11	禁 止				リザーブ	リザーブ
F0000H	ROM00	ROM01	ROM02	ROM03	ROM04	ROM05	リザーブ	リザーブ
E0000H								

● システムメモリエリア

DFFFFH	未使用	GVRAM	漢字 ROM #1	漢字 ROM #2 バックアップ ROM	辞書
B0000H	TVRAM				
A0000H					

● RAM エリア

9FFFFH	RAM バンク #0	RAM バンク #1	RAM バンク #2	RAM バンク ...
80000H	拡張メイン RAM			
40000H				
00000H				

3.2.1 RAM エリア

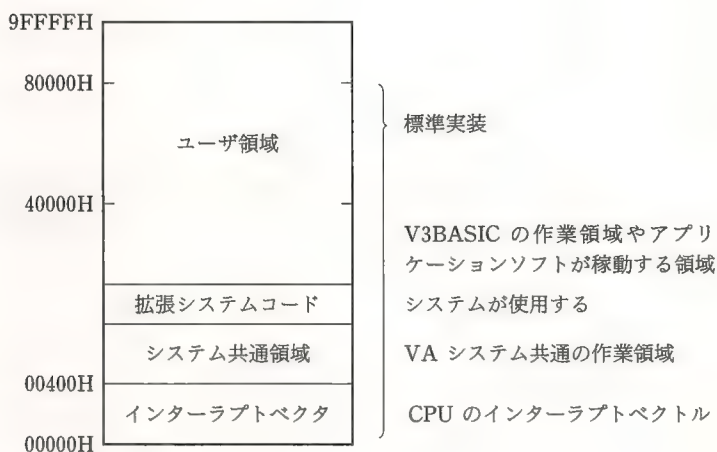
RAM エリアは次のように分かれる。

00000-3FFFFH	標準メイン RAM (本体内蔵)
40000-7FFFFH	拡張メイン RAM (拡張スロット)
80000-9FFFFH	バンク RAM (バンク #1 以降, 拡張スロット)

このうち 00000-7FFFFH は標準実装されている。

● メイン RAM

標準メイン RAM と拡張メイン RAM を合わせてメイン RAM といい、下図のようにマッピングされている。



インターラプトベクタ

CPU の割込みベクタ (256 個分)

システム共通領域

システム共通領域は VA システムで共通に使用される情報保持領域であり、アドレスが保証されているため総てのプログラムで参照することができる。また、アプリケーションソフトから設定可能もしくは設定する必要のある領域が存在する。なお、いかなるプログラムもこの領域を破壊してはならない。この領域はセグメント 40H から始まる。

拡張システムコード

ROM に置かれなかったコードやユーザ辞書・各種バッファ・非標準デバイスドライバなどが配置される。

● バンク RAM

バンク RAM は RAM バンク制御ポートによって 80000-9FFFFH のメイン RAM の代わりに CPU からアクセスできるようになる。その用途は特に決められていないが、RAM ディスクなどに利用できる。

3.2.2 システムメモリエリア

DFFFFH					
B0000H	未使用	GVRAM	漢字 ROM #1	漢字 ROM #2 バックアップ RAM	辞書
A0000H	TVRAM				

TVRAM

テキスト画面、スプライト表示用 RAM

GVRAM

グラフィック表示用 RAM

漢字 ROM

ANK 文字、漢字、ユーザ定義文字などのフォントが格納され、テキスト画面の表示に使われる。外字キャラクタージェネレータ用 RAM、メモリスイッチもこのエリアに入っている。

辞書 ROM

JFP に使われる辞書が入っている。

3.2.3 ROM エリア

● メイン ROM

標準実装されている ROM で、以下のように割り当てられている。

FFFFFH	ROM10	ROM11	禁 止			リザーブ	リザーブ
F0000H							
E0000H	ROM00	ROM01	ROM02	ROM03	ROM04	ROM05	リザーブ

各 ROM は主に次のような内容になっているが、将来にわたって確定したものではない。

ROM10	初期化ルーチン、V1/V2 スーパーバイザ、ファイルシステム テキスト・サウンド・キーボード・スプライト BIOS
ROM11	V1/V2 BASIC
ROM00	V3 BASIC、10 進演算パッケージ
ROM01	シェル、スクリーンエディタ、ファンシーフォント BIOS テキストエディタ、ポップアップ BIOS、ラインエディタ V1/V2 機械語モニタ
ROM02	日本語フロントプロセッサ、辞書アクセスルーチン セットアップ、アニメーション BIOS、マウス BIOS フロッピーディスク BIOS
ROM03	拡張グラフィック BIOS の一部、ハードディスク BIOS プリンタ BIOS、カレンダー時計 BIOS、コミュニケーション BIOS
ROM04	拡張グラフィック BIOS
ROM05	辞書の一部、デバッグ(V3)、グラフィック画面制御 BIOS

● 拡張 ROM

拡張スロットで拡張可能な ROM で、使用方法は決っていない。

3.3 メモリコントロール

各メモリ領域のバンクを選択するためにはメモリマップレジスタ (PORT 152H, 153H) または RAM バンク制御ポート (PORT 1D0H) を使用する。

● メモリモードの切換え(メモリマップレジスタ)

ポート 153H

ビット 6	システムメモリモード
0	V1/V2 モード
1	V3 モード(リセット時)

● ROM エリアのバンク切換え(メモリマップレジスタ)

ROM エリアは E0000-EFFFFH の ROM エリア 0 と F0000-FFFFFH の ROM エリア 1 とに分けられ、各々独立にバンク制御が可能である。

ポート 152H

<ROM エリア 0>		E0000-EFFFFH		
bit3	bit2	bit1	bit0	
0	0	0	0	ROM00 選択
0	0	0	1	ROM01 選択
0	0	1	0	ROM02 選択
0	0	1	1	ROM03 選択
0	1	0	0	ROM04 選択
0	1	0	1	ROM05 選択
0	1	1	0	リザーブ
0	1	1	1	リザーブ
1	×	×	×	バススロットに解放

<ROM エリア 1>		F0000-FFFFFH		
bit7	bit6	bit5	bit4	
0	0	0	0	ROM10 選択
0	0	0	1	ROM11 選択
0	0	1	0	禁止
0	0	1	1	禁止
0	1	0	0	禁止
0	1	0	1	禁止
0	1	1	0	リザーブ
0	1	1	1	リザーブ
1	×	×	×	バススロットに解放

● システムメモリエリアのバンク切換え(メモリマップレジスタ)

ポート 153H

bit3	bit2	bit1	bit0	
0	0	0	0	バススロット 選択
0	0	0	1	TVRAM 選択
0	0	1	0	リザーブ
0	0	1	1	リザーブ
0	1	0	0	GVRAM 選択
0	1	0	1	リザーブ
0	1	1	0	リザーブ
0	1	1	1	リザーブ
1	0	0	0	漢字 ROM #1 選択
1	0	0	1	漢字 ROM #2 選択
1	0	1	0	リザーブ
1	0	1	1	リザーブ
1	1	0	0	辞書 ROM 選択
1	1	0	1	リザーブ
1	1	1	0	リザーブ
1	1	1	1	リザーブ

(リザーブの部分はバススロット可)

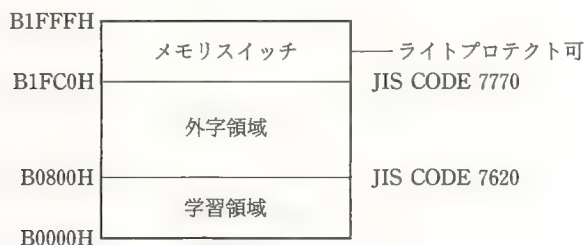
● RAM エリアのバンク切換え(RAM バンク制御ポート)

RAM エリアでバススロットに拡張できるのは 40000-9FFFFH の部分のみである。このうち、80000-9FFFFH をバンクエリアとして設定している。バンクを選択するためには、RAM バンク制御ポート (1D0H) に選択するバンク番号を出力する。

3.4 バックアップメモリ

漢字 ROM #2 の一部には 8 K バイトのバッテリーバックアップされた RAM があり，外字キャラクタージェネレータ，JFP 学習機能用領域，メモリスイッチとして使用されている。またメモリスイッチ領域にはライトプロテクトをかけることができる。

● バックアップメモリマップ



ライトプロテクト方法 (出力するデータはダミー)

OUT 198H	プロテクト ON
OUT 19AH	プロテクト OFF

メモリスイッチのうち 2 バイトは I/O ポートからもリードのみ可能である。

IN 30H	B1FC2H	番地リード
IN 31H	B1FC6H	番地リード

なお，バックアップメモリとしてのアクセスは V3 モードでのみ可能。ただし，バイトアクセスしかできない。

3.5 ウェイト

CPU が各メモリ領域をアクセスする場合、自動的にウェイトサイクルが挿入される場合がある。

標準メイン RAM	ウェイトなし
拡張メイン RAM	1 ウェイト
バンク RAM	1 ウェイト
TVRAM	最低 2 ウェイト 通常 7～13
GVRAM	最低 1 ウェイト
漢字 ROM	最低 2 ウェイト 通常 7～13
辞書 ROM	1 ウェイト
メイン ROM	ウェイトなし
拡張 ROM	1 ウェイト

V1/V2 モードではすべての領域で最低 1 ウェイト挿入される。

● TVRAM に対するアクセス時のウェイト

CPU や SGP (描画プロセッサ) が TVRAM やキャラクタジェネレータに対してアクセスを行う場合には、CPU(SGP)に対してウェイトサイクルが挿入される。

また 1 水平サイクル中のアクセス回数は使用している表示モード、スプライトの数等により制限があり、この制限を越える回数のアクセスを行った場合には、テキスト画面／スプライト画面の表示が正常に行われなくなる。

これを防止するために CPU(SGP) の TVRAM／キャラクタジェネレータに対する 1 水平サイクル中のアクセス回数を制限する機能があり、あらかじめ設定された回数以上のアクセスを行った場合には CPU(SGP) は次の水平サイクルにはいるまでウェイトさせられる。

制限回数の設定はポート 148H のビット 7～4(VALT2-0)で行う。

設定値	アクセス可能回数
0	制限なし
1	4 回
2	8 回
3	12 回
4	16 回
5	20 回
6	24 回
7	28 回

3.6 拡張スロットで拡張できるメモリ

VA で拡張スロットを使って拡張可能なメモリ空間(バンク)には以下のものがある。

ROM エリア 1	(F0000-FFFFFH)	128K バイト×8 バンク
ROM エリア 0	(E0000-EFFFFH)	128K バイト×8 バンク
システムメモリエリア	(A0000-DFFFFH)	256K バイト×11 バンク
拡張メイン RAM	(40000-9FFFFH)	384K バイト×1 バンク
バンク RAM	(80000-9FFFFH)	128K バイト×255 バンク

システムメモリエリアのうち 10 バンクはシステム予約であり使用できない。

● オートスタート ROM

ROM エリア 1 にマップされた拡張スロット上の ROM プログラムをシステム起動時にオートスタートさせることができる。これは、CPU からの特定のポートアクセスに対して、拡張ボード側がオートスタート要求フラグを出すことによって実現する。

	7	6	5	4	3	2	1	0
IN 156H	RBS7	RBS6	RBS5	RBS4	RBS3	RBS2	RBS1	RBS0

RBSn 0 = バンク n に ROM が実装されていて、オートスタートを要求している。

1 = ROM はない／オートスタートの必要がない。

[ROM ボード側の動作]

156H ポートリード時、設定したいバンクに相当する RBSn ビットが 0 になるようオープンコレクタゲートによりドライブする。その後、設定したバンクがポート 152H ビット 7~4 で指定されたならば ROM アクセス可能とする。

[CPU 側の動作]

システムイニシャライズ終了後、ポート 156H をリードし、0 となっているビットに相当するバンクを選択して F0000H 番地をコールする。複数のビットが 0 となっている場合には RBS0 から順に優先させる。

3.7 DMA によるメモリアクセス(V3 モード)

DMA によるメモリアクセスアドレスは CPU とほぼ同じであるが、メモリアクセス時のウェイトは、メイン RAM/ROM へのアクセスにウェイトがかかる場合がある。

バックアップメモリに対しての DMA アクセスはできない。

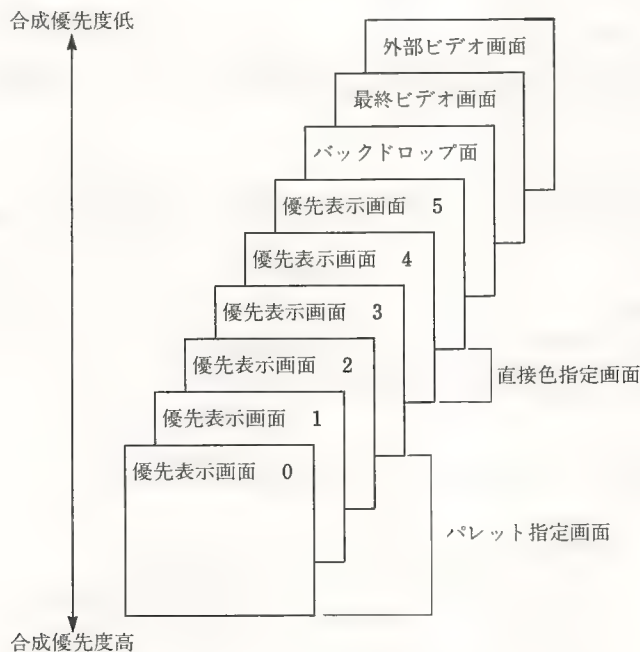
第4章 表示システム

4.1 表示システム概要

CRT 画面上には、テキスト／スプライト／グラフィック画面 0／グラフィック画面 1 の 4 種類の画像ソースとバックドロップ／外部ビデオが合成されて表示される。

4.1.1 画面構成

CRT 画面に表示される画面は以下のように構成される。



表示画面は上図のように 9 種の優先表示画面が合成されている。合成時の優先順位は優先表示画面 0 が最も高く、優先表示画面 1, 2 と順次低くなっていく。優先順位の高い画面ほど手前に見え、優先順位が低い画面は高い画面に隠されるように表示される。

優先表示画面 0～5 にはテキスト／スプライト／2 枚のグラフィックスの 4 つの画像ソースを割り当てる。

(1) パレット指定画面(優先表示画面 0～3)

優先表示画面 0～3 に割り付けられた画面は、パレット指定画面として表示される。パレット指定画面とは、ピクセルデータが直接表示色を指定するのではなく、ピクセルデータをカラーコードとして、そのカラーコードに対応するカラーパレットから得られた色によりそのピクセルの表示色が指定される画面である。

この画面に割り当てることができる画像ソースには以下のものがある。

- テキスト画面
- スプライト画面
- グラフィック画面 0(1ビット／ピクセル, 4ビット／ピクセル, 8ビット／ピクセル)
- グラフィック画面 1(1ビット／ピクセル, 4ビット／ピクセル, 8ビット／ピクセル)

(2) 直接色指定画面(優先表示画面 4, 5)

優先表示画面 4/5 に割り付けられた画面は、直接色指定画面として表示される。直接色指定画面とは、ピクセルデータによって直接表示色が決定される画面であり、カラーパレットの参照は行われない。

この画面に割り当てることができる画像ソースには以下のものがある。

- グラフィック画面 0(8ビット／ピクセル, 16ビット／ピクセル)
- グラフィック画面 1(8ビット／ピクセル)

(3) バックドロップ面

バックドロップ面は優先表示画面 5 の次の優先順位をもち、ボーダー領域を含んでバックドロップ／ボーダーカラーを表示する。

(4) 最終ビデオ画面

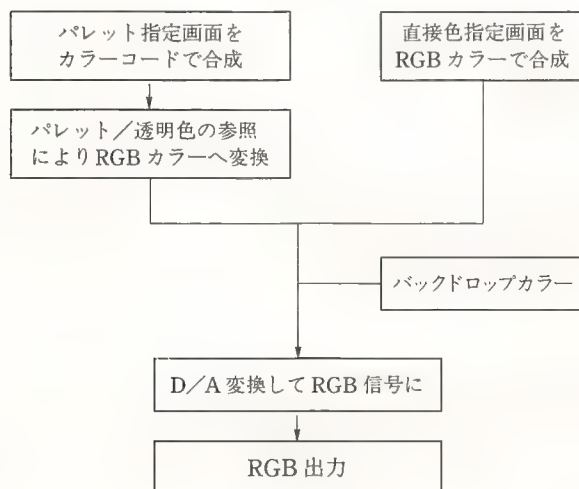
最終ビデオ画面は常に黒色となっており、上記のすべての画面が透明色の場合に黒レベルを提供する。

(5) 外部ビデオ画面

外部ビデオ画面は外部機器からのビデオ信号(TV 信号, VTR 信号等)とのスーパーインポーズを行う場合に外部ビデオ信号が表示される画面である。

●画面合成

各優先表示画面は以下のようなプロセスで合成される。



4.1.2 優先表示画面の指定

各優先表示画面 0～5 に使用の有無およびどの画像ソースを割り当てるかを指定する。グラフィック画面 0 / 1 はパレット指定画面(優先表示画面 0～3)および直接色指定画面(優先表示画面 4 / 5)のいずれにも割り当てることができるが、同時にパレット指定画面と直接色指定画面に割り当てると正常に表示されない。

●パレット指定画面の割当て指定

ワードポート 106H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
優先表示画面 3				優先表示画面 2				優先表示画面 1				優先表示画面 0			

〈各ニブルの値〉	〈指定画像ソース〉
0	指定なし(表示 OFF)
8	テキスト画面
9	スプライト画面
A	グラフィック画面 0
B	グラフィック画面 1

●直接色指定画面の割当て指定

ワードポート 108H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	優先表示画面 5				優先表示画面 4			

〈各ニブルの値〉	〈指定画像ソース〉
0	指定なし(表示 OFF)
8	グラフィック画面 0
9	グラフィック画面 1

4.1.3 テキスト画面とスプライト画面との関係

(1) 優先順位設定の制限

テキスト画面とスプライト画面は同一のテキストコントローラ(TSP)内で処理されており、スプライト画面はテキスト画面に対して常に高優先画面として出力される。したがって、優先表示画面の設定の際には必ずテキスト画面よりもスプライト画面の優先順位が高くなるよう設定する必要がある。

(2) テキスト画面とスプライト画面の分離

上述のようにテキスト画面とスプライト画面は同一のコントローラ内で処理されているのでコントローラからは 1 つの信号として出力される。したがって、テキスト画面とスプライト画面との間にグラフィック画面を挟み込むためには 2 つの画面を分離する必要がある。分離はテキスト画面／スプライト画面で使用するカラーコードに制限をつけることによって実現する。

テキスト画面／スプライト画面で利用できるトータルの色数は、透明色を除いて 15 色である。この 15 色を 1～N と N+1～15 の 2 つに分離し、各々をスプライト画面とテキスト画面で利用できる色とする。つまりカラーコード 1～N はスプライト画面のデータとして、カラーコード N+1～15 はテキスト画面のデータとして処理する。カラーコード 0 は透明色であり、テキスト／スプライト共通に使用できる。N が 0 のときはスプライト画面使用不可、15 のときはテキスト画面使用不可となる。

分離する境界(N)はポート 111H で設定する。

ポート 111H

ビット 7～ビット 4：テキスト／スプライト判別境界カラー

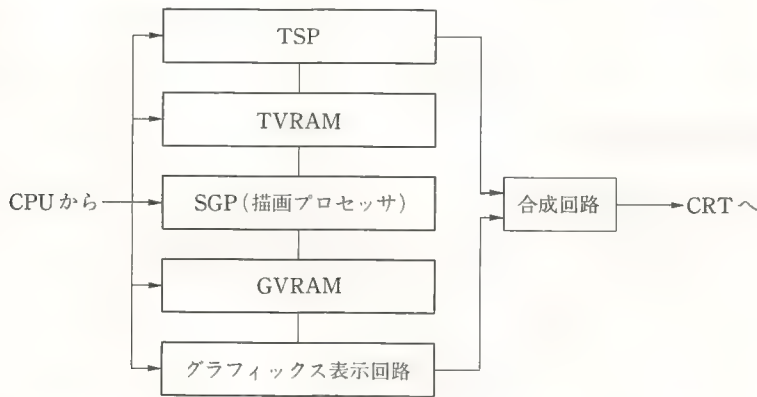
1～設定値 : スプライト

設定値 + 1～15：テキスト

なおテキスト画面とスプライト画面が隣合っているときにはこのポートは通常意味がない。またこの機能はテキスト画面とスプライト画面の分離であって、優先順位を変更するものではなく、常にスプライト画面のほうが高優先である。

4.1.4 表示制御回路の構成

VA の表示制御回路の構成の概要は次のようになっている。



各構成の役割

TSP + TVRAM : テキスト／スプライト表示

グラフィックス表示回路 + GVRAM : グラフィックス表示

SGP : VRAM への描画

4.2 テキスト表示機能

テキスト表示回路はキャラクタジェネレータに登録されている文字フォントを TVRAM に格納されているハードウェア文字コードにより指定し、同時に TVRAM に格納されているアトリビュートコードにより文字単位に修飾して表示する。

テキスト表示機能には、V3 対応モードと V1/V2 対応モードとがあり、V1/V2 対応モードにおいては 88M/F シリーズと互換性のあるテキスト表示機能が提供される。

V3 対応モードは以下の条件により設定される。

ポート 148H

ビット 3(ATM)=0 : アトリビュートモードが V3 モード

ビット 1(TVWM)=1 : TVRAM をワード構成にする

TSP

ネイティブモード(通常モード)

V3 対応モードでは TVRAM はワード構成となっており、1 ワードの文字コードにより半角文字が表示される。表示される文字のレターフェースは 8×16 ドットが標準となり、これを左右のペアで使用するによって全角の漢字の表示を行う。

表示文字のボディフェースは横方向が 8 ドット固定で縦方向が 16 ドット～32 ドットを指定することができる。

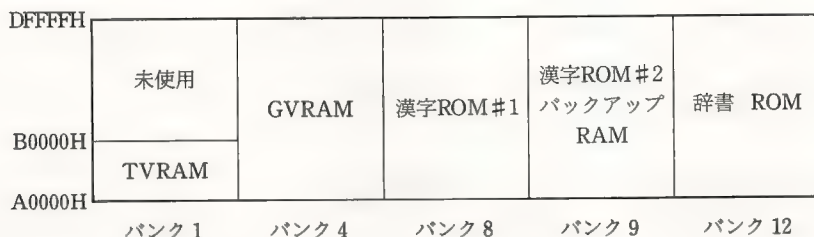
全角漢字の表示文字数は最大で、200 ライン時には 40 文字×12 行、400 ライン時には 40 文字×25 行となる。

4.2.1 テキスト VRAM(TVRAM)

(1) CPU アドレス

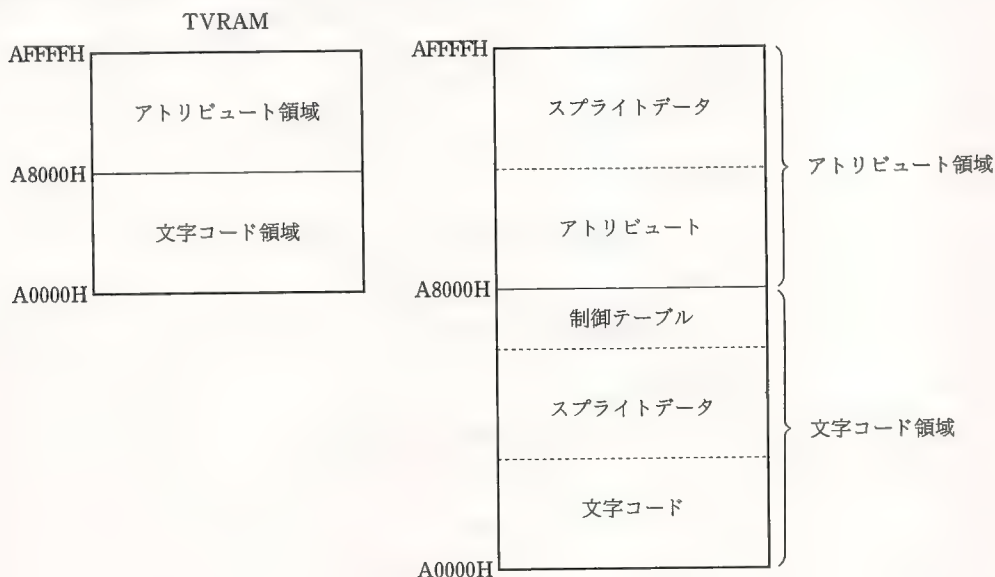
TVRAM は CPU メモリ空間の A0000-AFFFFH に割り当てられており、システムメモリエリアのバンク #1 となっている。システムメモリエリアのバンク選択にはワードポート 152H のビット 11～8 を用いる。

●システムメモリエリア



容量としては全部で 64K バイトがあるが、32K バイトずつの文字コード領域とアトリビュート領域の 2 つに分けられている。

文字コードやアトリビュートは各ブロックの先頭から必要なだけ使用され、残りはテキスト／スプライト制御テーブルやスプライトデータが置かれる。



(2) TSP アドレス

テキストコントローラである TSP から見た TVRAM はワードごとにアドレスされており、そのアドレスは、

文字コード領域……………0000～3FFFH
アトリビュート領域……8000～BFFFH

である。TSP に与えるパラメータではこのアドレスを使用する場合と CPU アドレス(オフセット)を使用する場合があるので注意が必要である。

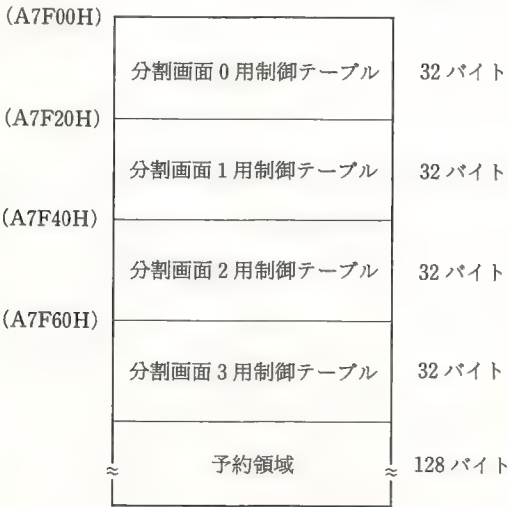
CPU アドレスと TSP アドレスの対応は次のようになっている。

TSP アドレス(16進)	CPU アドレス(16進)
0000	0001／ 0000
0001	0003／ 0002
⋮	⋮
3FFF	7FFF／ 7FFE
8000	8001／ 8000
8001	8003／ 8002
⋮	⋮
BFFF	FFFF／ FFFE

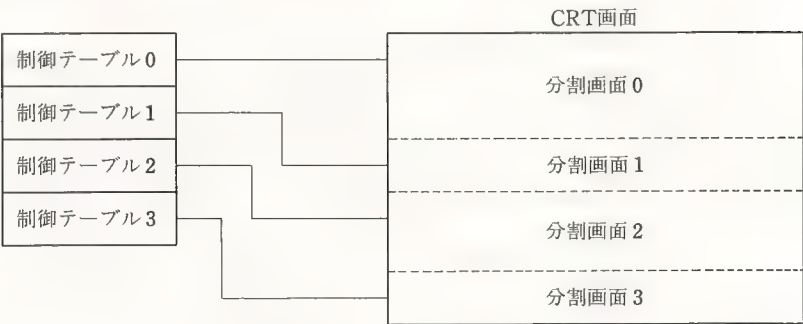
4.2.2 テキスト表示フレームバッファの設定

テキスト表示は上記の文字コード領域内に表示用フレームバッファが設定され、そのフレームバッファ内に文字コードが格納される。アトリビュートコードは文字コードのアドレス+8000H に格納されているデータが使用される。

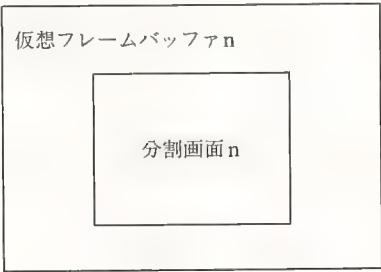
テキスト表示フレームバッファの設定は TVRAM 中にあるテキスト画面制御テーブルによって行う。画面制御テーブルの位置は TSP へ設定することにより TVRAM 中の任意のアドレスに設定できるが、通常は CPU アドレスで A7F00H を使用する。テーブルとして 256 バイトの領域を割り当ててあるが、実際にはそのうち前半の 128 バイトを使用する。



前図のように制御テーブルは4つあり、CRT画面を水平に4分割して使用することができる。ただし、分割画面の間に隙間をつくることはできない。



分割画面は仮想的なフレームバッファの一部として捉えられ、フレームバッファの中を自由にスムーズスクロールできる。ただし、フレームバッファからはみだしたときのラップアラウンドは行われない。



TVRAM と分割画面との対応は次のようになっている。分割画面の先頭アドレスからのオフセットを示す。

0	2	4	6	8	⋮	DW-2
VW	VW+2	VW+4	VW+6			VW+DW-2
VW * 2	VW * 2+2	VW * 2+4	VW * 2+6			
VW * 3	VW * 3+2	VW * 3+4				
VW * 4	VW * 4+2					
VW * 5						
⋮						⋮
VW*(DH-1)						VW*(DH-1)+DW-2

VW はフレームバッファの横幅(バイト)
DW は分割画面の横幅(バイト=文字数×2)
DH は分割画面の行数(バイト)

各画面制御テーブルにはフレームバッファと分割画面を定義する。その構成は次のようになっている。

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
+00	フレームバッファスタートアドレス(VSA)															
+02	Reserved								0	0	0	0	0	VSA の上位		
+04	0	0	0	0	0	フレームバッファの高さ(VH)										
+06	Reserved															
+08	0	0	0	0	0	0	フレームバッファの横幅(VW)									
+0A	バックグラウンドカラー				フォアグラウンドカラー				Reserved				表示モード			
+0C	0	0	0	ラストアドレスオフセット				Reserved								
+0E	Reserved															
+10	分割画面スタートアドレス(RSA)															
+12	Reserved								0	0	0	0	0	0	0	0
+14	0	0	0	0	0	0	0	分割画面の高さ(RH)								
+16	0	0	0	0	0	0	分割画面の横幅(RW)									
+18	0	0	0	0	0	0	0	分割画面の垂直開始位置(RYP)								
+1A	0	0	0	0	0	0	分割画面の水平開始位置(RXP)									
+1C	Reserved															
+1E	Reserved															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

①フレームバッファスタートアドレス

フレームバッファの開始アドレスを指定する。

TVRAM の先頭 (A0000H) を 0 としたバイトアドレスで設定する。

偶数アドレスしか設定できない。

②フレームバッファの高さ

フレームバッファの高さ (縦幅) を行数で設定する。

③フレームバッファの横幅

フレームバッファの横幅をバイト数で設定する (1 文字 2 バイト)。

④バックグラウンド／フォアグラウンドカラー

表示文字のバックグラウンドとフォアグラウンドのパレット番号を指定する。

表示モードによってはどちらかまたは両方を使用しないこともある。

⑤表示モード

アトリビュートの付け方のモードを指定する。

0～5 を使用する (4.2.4 参照)。

⑥ラストアドレスオフセット

垂直スムーズスクロール用。

設定したラスタ数だけ (最大 1 行分) 表示画面が上にスクロールする。

⑦分割画面スタートアドレス

文字コード領域中の表示する部分 (分割画面) の先頭アドレスを指定する。

TVRAM の先頭 (A0000H) を 0 としたバイトアドレスで設定する。

偶数アドレスしか設定できない。

⑧分割画面の高さ

分割画面の高さ (縦幅) をラスタ数で指定する。4 つの分割画面の高さの合計は前画面の高さに一致させる必要がある。16 以上の偶数しか設定できない。

⑨分割画面の横幅

分割画面の横幅をドット数で指定する。

32 の倍数しか設定できない。(設定値 / 8 + 2) 文字が表示される。

⑩分割画面の垂直開始位置

分割画面の CRT 上での垂直開始位置をラスク単位で指定する。前の分割画面に連続するように設定する。偶数しか設定できない。

⑪分割画面の水平開始位置

水平スムーズスクロール用

分割画面の CRT 上での水平開始位置をドット単位で指定する。
負の数も使用できる。

4.2.3 文字コード

文字コードとしては 2 バイトのハードウェア文字コードが使用されている。全角文字を表示する場合には文字の左右のペアを使用する。

● 2 バイト文字(日本語文字)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
JIS 第二バイト								0	JIS 第一バイトー20H						
0 : 文字の左側 1 : 文字の右側															

● 1 バイト文字(ANK 文字)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	1 バイトコード							

4.2.4 アトリビュートコード

アトリビュートは文字コードのアドレスに TSP の DSPDEF コマンド(4.4.2 参照)で設定された値(通常 8000H)を足したアドレスにあるデータが使用される。各文字につき 1 ワード割当てであるが、上位バイトは現在使用しておらず、0 を設定する。

アトリビュートではフォアグラウンド/バックグラウンドの色指定とブリンク/リバースなどの形状指定を行うことができる。

色指定ではフォアグラウンド/バックグラウンドに使用するパレット番号を指定する。フォアグラウンドカラーは 16 色から、バックグラウンドでは 16 色または 8 色から選択する。8 色から選択する場合はパレット 0 ~ パレット 7 から指定する。

形状指定としては全部で次の 7 種類があり、モードによって使用できるものと使用できないものがある。水平ラインの位置は TSP の DSPDEF コマンドで設定する。

SECRET	シークレット(1でON)	文字表示を行わない
BLINK	ブリンク(1でON)	文字をブリンクする
REVRSE	リバース(1でON)	文字/地を反転する
HLINE	水平ライン(1でON)	文字に水平線を引く
ULINE	アンダーライン(1でON)	文字に下線を引く
DWID	ダブルウィドス(1でON)	文字の半分を横倍する
DWIDC	DWID コントロール(1で右側)	DWID で横倍する左右を指定する

アトリビュートには 6 種のモードがあり、画面制御テーブルの表示モードフィールドで指定する。このモードは各分割画面ごとに独立して設定できる。

(1) アトリビュートモード0(表示モード=0)

各文字単位に独立にフォアグラウンドカラー及びバックグラウンドカラーを1色/16色指定する。画面制御テーブルのフォアグラウンドカラー・バックグラウンドカラーは使用されない。また形状指定はできない。

7	6	5	4	3	2	1	0
バックグラウンドカラー				フォアグラウンドカラー			

(2) アトリビュートモード1(表示モード=1)

各文字単位にフォアグラウンドカラーを1色/16色、形状指定としてシークレット/プリנק/リパス/ホリゾンタルラインの組合せを指定する。バックグラウンドカラーは画面制御テーブルのバックグラウンドカラーフィールドで設定した色が使用される。

7	6	5	4	3	2	1	0
フォアグラウンドカラー				HLINE	REVRSE	BLINK	SECRET

(3) アトリビュートモード2(表示モード=2)

各文字単位シークレット/プリנק/リパス/ホリゾンタルライン/アンダーライン/ダブルウィドス/ダブルウィドスコントロールの形状指定の組合せを設定する。色指定としては画面制御テーブルで設定したフォアグラウンドカラーおよびバックグラウンドカラーが使用される。

7	6	5	4	3	2	1	0
DWIDC	DWID	ULINE	HLINE	—	REVRSE	BLINK	SECRET

(4) アトリビュートモード3(表示モード=3)

アトリビュート指定としてカラー指定と形状指定があり、アトリビュートコードのビット3で区別する。カラー指定ではフォアグラウンドカラーを1色/16色、バックグラウンドカラーを1色/8色指定する。また形状指定ではシークレット/プリנק/リパス/ホリゾンタルライン/アンダーライン/ダブルウィドス/ダブルウィドスコントロールを指定する。設定されたアトリビュートは再度同種のアトリビュートが設定されるまで使用される。また1文字でカラー指定と形状指定とを同時に変更することはできない。画面制御テーブルのフォアグラウンドカラー・バックグラウンドカラーは使用されない。

7	6	5	4	3	2	1	0
フォアグラウンドカラー				1	バックグラウンドカラー		

7	6	5	4	3	2	1	0
DWIDC	DWID	ULINE	HLINE	0	REVRSE	BLINK	SECRET

(5) アトリビュートモード4(表示モード=4)

各文字単位カラー指定としてフォアグラウンドカラーを1色/16色、バックグラウンドカラーを1色/8色指定でき、形状指定としてプリנקが指定できる。画面制御テーブルのフォアグラウンドカラー・バックグラウンドカラーは使用されない。

7	6	5	4	3	2	1	0
BLINK	バックグラウンドカラー			フォアグラウンドカラー			

(6)アトリビュートモード5(表示モード=5)

各文字単位にカラー指定としてフォアグラウンドカラーを1色/16色, バックグラウンドカラーを1色/8色指定でき, 形状指定としてプリンクと水平ラインが指定できる。フォアグラウンドカラーの1または9を使用するとカラー#1(または#9)+水平ラインで表示される。画面制御テーブルのフォアグラウンドカラー・バックグラウンドカラーは使用されない。

7	6	5	4	3	2	1	0
BLINK	バックグラウンドカラー			フォアグラウンドカラー			

下位3ビットが001のとき水平ライン

4.2.5 ANK文字の表示フォント

ANK文字フォントには8×8ドットと8×16ドットの2種類があり, 使用するCRTと用途により選択して使用する。8×8ドットフォント使用時には日本語表示はできない。この選択は次のポートで行う。

ポート148H

ビット2(ANKM)

0:横8ドット×縦16ドット(400ライン表示時)

1:横8ドット×縦8ドット(200ライン表示時)

4.2.6 40文字表示

40文字/行の表示は, ダブルウィドスアトリビュートではなくTSPの外部回路で実現している。その切換えは次のポートを用いる。

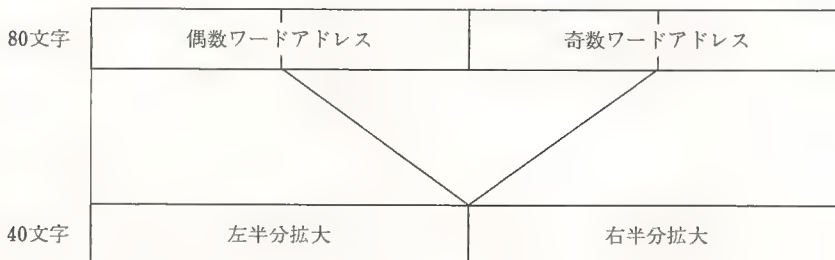
ポート30H

ビット0(80CM)

0:40文字/行

1:80文字/行

40文字/行モードが設定されるとキャラクタジェネレータから読み出されたフォントデータは水平方向に2倍に拡大されてTSPに入力される。フォントの拡大ルールはV3対応モードとV1/V2対応モードとは異なる。V3対応モードでは, ワードアドレスで偶数アドレスのフォントは左半分が拡大され, 奇数アドレスのフォントは右半分が拡大されて表示される。したがって, 偶数/奇数ワードアドレスには同一の文字コードを格納しておく必要がある。



4.2.7 テキスト表示禁止機能

(1) 分割画面の1つをブランクにするとき

表示制御テーブルの分割画面の横幅 = 0 に、水平開始位置 = 3F0H にする。

(2) 全テキスト表示を禁止する

ポート 148H のビット 7 = 1 にする。TSP の表示停止コマンドではスプライト表示も禁止されてしまう。

4.3 スプライト表示機能

TSP は最高 32 枚のスプライトを用いて動きのあるパターンも表示することができる。各スプライトには優先順位が付けられており、2 つ以上のスプライトが重なった場合の優先順位の低いスプライトの重なった部分は優先順位の高いスプライトによって隠されて表示されない。スプライトの優先順位はスプライト制御テーブル内の順序で決定される。またスプライトはテキストよりも常に優先順位が高い。色数として 16 色モードとモノクロモードがあり、いずれもパレットにより 4096 色中から選択できる。

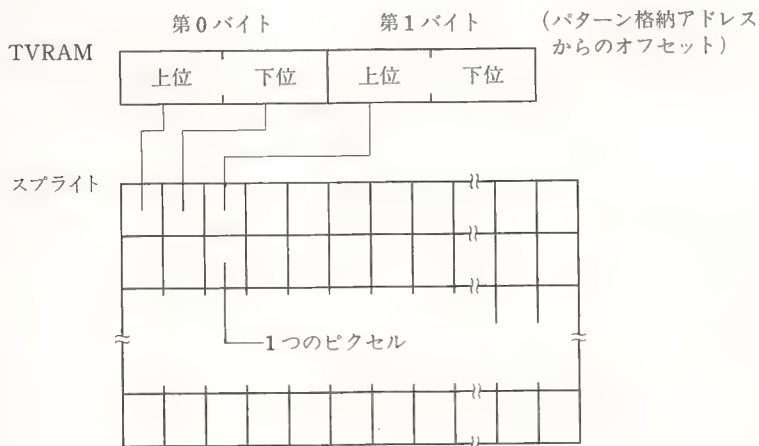
4.3.1 スプライトモードと VRAM

スプライト用の VRAM には 64K バイトの TVRAM のうちテキスト表示に使用していない部分をすべて用いることができる。ただし文字コード領域とアトリビュート領域の境界(A8000H 番地)にまたがってデータを置くことはできない。

スプライトには色数によって 2 つの表示モードがあり、各スプライトごとに独立に指定することができる。

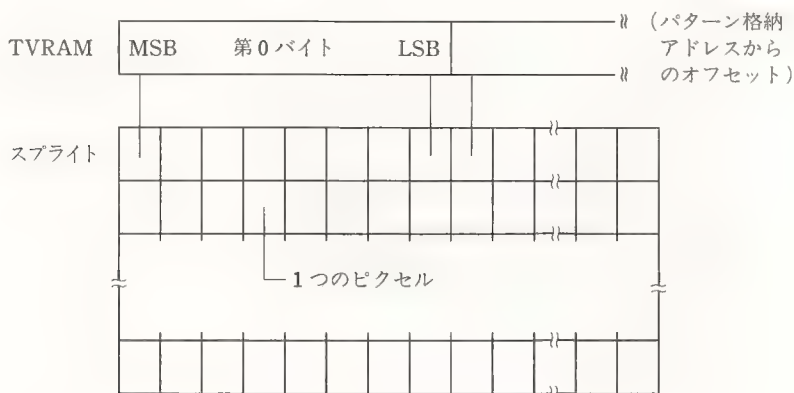
(1) スプライトモード 0(16 色モード)

ピクセル単位に色指定できるモードである。TVRAM 中のスプライトパターンデータの各ニブルに 0~15 のカラーコードを設定する。カラーコード 0 は常に透明色として扱われる。



(2) スプライトモード1(モノカラーモード)

スプライト単位に色指定できるモードである。スプライト表示パターンのうち、1となっているピクセルの色がスプライト制御テーブル(後述)で設定したフォアグラウンドカラーで表示され、0となっているピクセルは透明色またはカラーコード8で表示される。



4.3.2 スプライト制御テーブル

スプライトの大きさおよび表示位置などの設定は TVRAM 中にあるスプライト制御テーブルによって行う。スプライト制御テーブルの位置は TSP へ設定することにより TVRAM 中の任意のアドレスに設定できるが、通常は CPU アドレスで A7E00～A7EFFH を使用する。低いアドレスに制御テーブルがあるスプライトほど優先順位が高い。

(A7E00H)	スプライト 0 用制御テーブル	8 バイト (# 0 はマウスカーソル用)
(A7E08H)	スプライト 1 用制御テーブル	8 バイト
(A7E10H)	スプライト 2 用制御テーブル	8 バイト
(A7E18H)	スプライト 3 用制御テーブル	8 バイト
(A7E20H)		
≈		≈
(A7EF8H)	スプライト 31 用制御テーブル	8 バイト (# 31 はテキストカーソル用)

各スプライト制御テーブルの構成は次のようになっている。

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
+00	垂直サイズ (VSIZE)						SW	垂直表示位置 (YP)								
+02	水平サイズ (XSIZE)					MD	水平表示位置 (XP)									
+04	スプライトデータアドレス (SPDA)															
+06	0	0	0	0	0	0	0	フォアグランドカラー					BC	0	0	0

- ①垂直サイズ スプライトの垂直方向のサイズ
(ライン数/4-1)を設定する。4ライン～256ラインを4ラインごとに指定できる。
- ②垂直表示位置 CRT画面上の表示位置のY座標
Y座標(ライン位置)を設定する。負数も使用可能
- ③スプライトスイッチ(SW)
当該スプライトの表示指定。1で表示ON。
- ④水平サイズ スプライトの水平方向のサイズ
 - 16色モードのとき
水平ドット数/8-1を設定する。
8～256ドットを8ドットおきに指定できる。
 - モノクロモードのとき
水平ドット数/32-1を設定する。
32～1024ドットを32ドットおきに指定できる。
- ⑤水平表示位置 CRT画面上の表示位置のX座標
X座標(ドット位置)を設定する。負数も使用可能。
スプライトの水平解像度は常に640ドットである。
- ⑥スプライトモード(MD)
 - 0で16色モード(スプライトモード0)
 - 1でモノクロモード(スプライトモード1)
- ⑦スプライトデータアドレス
スプライトのパターンデータが格納されているアドレス。
TSPアドレス(ワードアドレス)を指定する。
- ⑧フォアグラウンドカラー
モノクロモード時のフォアグラウンドカラー
- ⑨バックグラウンドカラー(BC)
モノクロモード時のバックグラウンドカラー
0で透明色, 1でカラーコード#8

4.3.3 スプライトの表示能力

スプライトの大きさや表示における制限などについて述べる。

(1) スプライトのサイズ

表示できるスプライトのサイズはスプライトモードによって異なる。

- 16色モード(スプライトモード0)
 - 水平サイズ: 8～256ドット 8ドット単位で指定
 - 垂直サイズ: 4～256ドット 4ドット単位で指定
- モノクロモード(スプライトモード1)
 - 水平サイズ: 32～256ドット 32ドット単位で指定
 - 垂直サイズ: 4～256ドット 4ドット単位で指定

(2) スプライトの数と優先順位

スプライトは最大32個を設定/表示することができる。複数のスプライトが重なった場合の優先順位はスプライト制御テーブルが低いアドレスにあるスプライトほど優先順位が高い。またテキスト画面に対してスプライトは常に優先順位が高い。

1ラスタ上に(重ならなくても)同時に表示できるスプライト数は、数自体では制限を受けないが、1ラスタ上の合計データ数によって制限される。下にあげたメモリアクセスの合計が1ラスタ

の表示時間を超えた場合には正常な表示が行われなくなる。通常、16色モードで横2百数十ドットまで可能である。

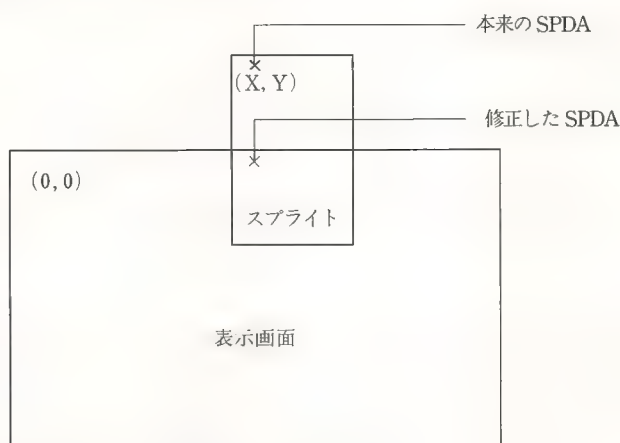
- セットアップ／リフレッシュのためのオーバーヘッドタイム
- スプライト制御テーブルへのアクセス
- スプライトパターンデータへのアクセス
- テキストデータへのアクセス
- CPU／SGP からの TVRAM アクセス
- TSP のコマンド処理のためのメモリアccess

(3) スプライトの表示位置

水平方向(X座標)：0～1023(−1は1023と同じ)

垂直方向(Y座標)：0～511(−1は511と同じ)

スプライトが画面の左右または下方にはみ出す場合には表示位置を設定だけで自動的にクリッピングされるが、上方にはみ出す場合には表示位置だけでなく、スプライトデータアドレスも変更しなければならない。スプライトパターンの先頭アドレスではなく、スプライトパターン中で画面に表示される部分の先頭アドレスを設定する。垂直サイズは変更しなくてよい。



(4) スプライトの解像度

スプライト画面の横方向の表示解像度は常に640ドット、縦方向の表示解像度は200(204)／400(408)である。ただし、インターレースモード1のときテキスト／グラフィックスと異なり400ライン表示はできない。

(5) 縦ズーム

TSPのSPRONコマンド(4.4.2参照)でスプライトの縦ズーム機能を指定できる。これは200ライン用のスプライトデータを400ライン表示で使用するときに使われる。

4.3.4 スプライトの衝突検出機能

スプライトどうしが表示画面内で重なり合った場合、これをスプライトの衝突として検出することができる。また、スプライトを2つのグループに分けてグループ間衝突を検出することができる。

(1) 衝突検出フラグ

スプライトの衝突が検出された場合には TSP ステータスのスプライトコントロールフラグがセットされる。このフラグは TSP へのコマンドによって1ラスタ中のスプライト数を制限した場合に、その制限を超えるスプライトが1ラスタ中に存在した時にもセットされる。このフラグは VRTC ごとに更新される。

● TSP ステータス(IN 142H)

7	6	5	4	3	2	1	0
LP	VB	SC	ER	EMEN	BUSY	OBF	IBF

└────────── スプライトコントロールフラグ

(2) 衝突検出条件

スプライトの衝突検出は以下の条件において行われる。

- ①衝突検出は有効表示画面内だけで行われる。
- ②衝突は両スプライトの表示パターンの不透明色どうしが重なったことによって検出される。
- ③衝突グルーピングモードによりグルーピングされている場合には、異種のグループ間でのみ衝突が検出される。
- ④1回の画面表示において最初に検出された衝突により衝突フラグがセットされるため1画面中では1箇所の衝突のみが検出される。

(3) 衝突グルーピングモード

スプライトを2つのグループに分けて衝突検出することができる。

(3.1) 衝突グルーピングモード0

スプライトのグルーピングを行わないモードであり、全スプライトに対して一律に衝突検出を行う。

(3.2) 衝突グルーピングモード1

カラーコードによってスプライトを2つのグループに分け、グループ間の衝突検出を行う。同じグループ内での衝突検出は行われない。

グルーピングは各ピクセルの表示色(カラーコード)の最上位ビットによって行われる。したがって、スプライトモード0(16色モード)では各ピクセル単位に、スプライトモード1(モノクロモード)ではスプライト単位にグルーピングされる。

グループ0 カラーコード1～7のピクセル

グループ1 カラーコード8～15のピクセル

カラーコード0は透明色であり、衝突検出は行われない。

4.4 TSP(テキスト／スプライトプロセッサ)

テキスト画面とスプライト画面の制御に用いられている TSP について解説する。

TSP 自身の機能としては、テキスト／スプライト／セミグラフィックス／グラフィックス／3301 エミュレーションの 5 種がある。しかし、VA ではこの内テキストとスプライト、3301 エミュレーション機能だけを用い、3301 エミュレーション機能は V1/V2 モードで使用している。

TSP で使用するコマンドには次のようなものがある。

コマンド番号／名前		意 味
10H	SYNC	TSP の初期設定を行う。表示は停止する。
12H	DSPON	表示を開始する。
13H	DSPOFF	表示を停止する。
14H	DSPDEF	画面構成を設定する。
15H	CURDEF	カーソルを定義する。
16H	ACTSCR	カーソルを置く分割画面を指定する。
1EH	CURS	カーソルの位置を指定する。
8CH	EMUL	3301 エミュレーションを開始する。
88H	EXIT	コマンドを中断する。エミュレーションを停止する。
82H	SPRON	スプライトの表示を開始する。
83H	SPROFF	スプライトの表示を停止する。
85H	SPRSW	個別にスプライトの表示を ON/OFF する。
81H	SPROV	スプライトオーバー／衝突の情報を獲得する。

4.4.1 コマンドの与え方

TSP へは I/O ポートからコマンドを与えて制御する。1 つのコマンドを設定するには、まず 1 バイトのコマンドを与え、続いて必要なバイト数だけパラメータを与える。

TSP のポートアドレスは次のようになっている。

142H コマンドバイト書込み／ステータス読出し

146H パラメータ入出力

TSP へコマンドを与える時にはステータスのビット 0(Input buffer full bit)とビット 2(Busy bit)を見る必要がある。

コマンドバイト／パラメータを設定するルーチンの例を示す。

<コマンドバイト設定ルーチン>

```

MOV      AH, AL          ; save COMMAND BYTE
MOV      DX, 142H        ; STATUS port adrs
NOT_READY:
IN        AL, DX          ; read STATUS
TEST     AL, 5           ; bit0=IBF, bit2=BUSY
JNZ      NOT_READY       ; Loop if not ready
MOV      AL, AH          ; restore COMMAND BYTE
OUT      DX, AL          ; write COMMAND BYTE
RET
```


〈パラメータ設定ルーチン〉

```

MOV     AH, AL           ; save PARAMETER
MOV     DX, 142H         ; STATUS port adrs
NOT_READY:
IN      AL, DX           ; read STATUS
TEST    AL, 5            ; bit0=IBF, bit2=BUSY
JNZ     NOT_READY        ; Loop if not ready
MOV     AL, AH           ; restore PARAMETER
MOV     DX, 146H         ; PARAMETER port adrs
OUT     DX, AL           ; write PARAMETER
RET

```

4.4.2 各コマンドの解説

(1) SYNC コマンド

TSP の初期設定をおこなう。垂直解像度を変えるときにはこのコマンドを再設定しなければならない。

コマンド 10H

パラメータ (14 バイト)

● 200 ライン

15.98KHz ノンインターレース : C1 57 1C 00 9F 00 10 0F 25 00 C8 00 0F 08

15.73KHz インターレース : C1 47 1C 00 9F 00 12 11 24 00 C8 00 17 04

24.8KHz ノンインターレース : 81 57 10 00 9F 00 10 0F 19 00 90 40 07 08

● 204 ライン

15.98KHz ノンインターレース : C1 57 1C 00 9F 00 10 0F 23 00 CC 00 0D 08

15.73KHz インターレース : C1 47 1C 00 9F 00 12 11 22 00 CC 00 15 04

24.8KHz ノンインターレース : 81 57 10 00 9F 00 10 0F 19 00 98 40 07 08

● 400 ライン

15.73KHz インターレース : 41 47 1C 00 9F 00 12 11 24 00 C8 00 17 04

24.8KHz ノンインターレース : 01 57 10 00 9F 00 10 0F 19 00 90 40 07 08
(スプライト 200 ライン)

C1 57 10 00 9F 00 10 0F 19 00 90 40 07 08
(スプライト 400 ライン)

● 408 ライン

15.73KHz インターレース : 41 47 1C 00 9F 00 12 11 22 00 CC 00 15 04

24.8KHz ノンインターレース : 01 57 10 00 9F 00 10 0F 19 00 98 40 07 08
(スプライト 200 ライン)

C1 57 10 00 9F 00 10 0F 19 00 98 40 07 08
(スプライト 400 ライン)

(2) DSPON コマンド

TSP の表示を開始する。

コマンド 12H

パラメータ (3 バイト)

- #1 テキスト画面制御テーブルのアドレス
- #2 0
- #3 0

[解説]

#1: TVRAM 64K バイト中のバイトアドレスの上位バイトを設定する。

(3) DSPOFF コマンド

TSP の表示を停止する。スプライトの表示も停止する。

コマンド 13H

パラメータ なし

(4) DSPDEF コマンド

画面の構成と表示形態を指定する。

コマンド 14H

パラメータ (6 バイト)

- #1 アトリビュートオフセット(下位バイト)
- #2 アトリビュートオフセット(上位バイト)
- #3 文字コードの格納ピッチ×16
- #4 文字ボディのラスタ数
- #5 ホリゾンタルラインの表示ラスタ位置
- #6 プリント速度

[解説]

#1/#2: 文字コード領域とアトリビュート領域の差のバイト数を設定する。

通常は 8000H である。

#3: 文字コードが何バイトおきに格納されているかを指定する。

通常は 20H(2 バイトおき)である。

#4: 1 行のラスタ数を指定する。最低 8 ラスタ。

実際のラスタ数(画面の全ラスタ数/表示行数)-1 を設定する。

#5: 行中でホリゾンタルラインを表示するラスタを設定する。

(設定値+1)ラスタ目に表示される。

#6: プリント文字およびカーソルの点滅速度を指定する。

プリント文字: 明=設定値×24 フレーム 暗=設定値×8 フレーム

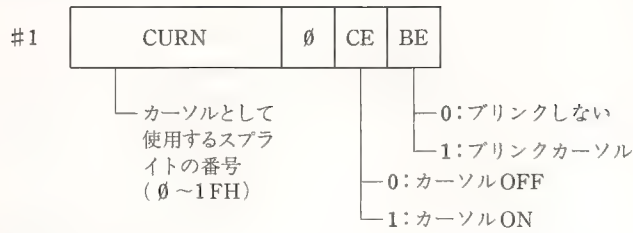
カーソル : 明=設定値×8 フレーム 暗=設定値×8 フレーム

(5) CURDEF コマンド

テキストカーソルの形態を指定する。カーソルはスプライトを利用しており、スプライト制御テーブルも設定する必要がある。

コマンド 15H

パラメータ (1 バイト)



(6) ACTSCR コマンド

カーソルを置く分割画面を指定する。

コマンド 16H
パラメータ (1 バイト)
#1 分割画面番号×32

(7) CURS コマンド

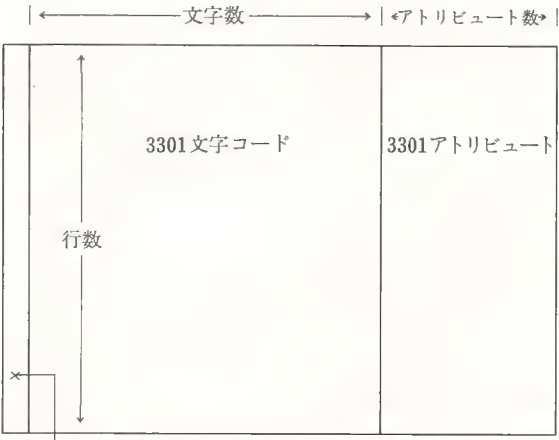
カーソルの位置を指定する。

コマンド 1EH
パラメータ (4 バイト)
#1 垂直位置(下位バイト)
#2 垂直位置(上位バイト)
#3 水平位置(下位バイト)
#4 水平位置(上位バイト)

(8) EMUL コマンド

3301 エミュレーションを開始する。3301 エミュレーションは TSP が 3301 用のアトリビュートから TSP アトリビュートへの展開を動的に行うことによって実現する。このコマンド実行後は、他のコマンドが入力されるまで展開を行いつづける。

コマンド 8CH
パラメータ (4 バイト)
#1 エミュレーションする分割画面番号×32
#2 文字数
#3 アトリビュート数
#4 行数



(9) EXIT コマンド

パラメータの受付途中にあるコマンド処理を中止してコマンド待ち状態にする。単に 3301 エミュレーション動作を中止するときにも用いる。

コマンド 88H

パラメータ なし

(10) SPRON コマンド

スプライトの表示を開始する。カーソル設定時にも必要となる。

コマンド 82H

パラメータ (3 バイト)

#1 スプライト制御テーブルのアドレス

#2 0

#3	HSPN	0	MG	GR
----	------	---	----	----

[解説]

#1 TVRAM 64K 中のバイトアドレスの上位バイトを設定する。

#3 HSPN 1 ラスタ中に同時に表示できるスプライトの最大数。
(設定値+1) より多い数のスプライトが 1 ラスタ中に存在するとスプライトオーバーとなる。

MG 1 のときすべてのスプライトを縦方向に 2 倍に拡大する。

GR グルーピングモード。1 でグルーピング衝突検出する。

(11) SPROFF コマンド

スプライトの表示を停止する。カーソルの表示も行われない。

コマンド 83H

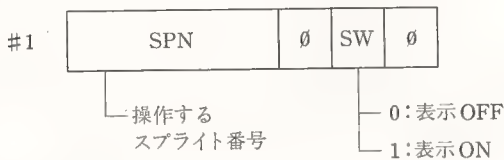
パラメータ なし

(12) SPRSW コマンド

各スプライトを個別に ON/OFF する。このコマンドを使用せず、直接スプライト制御テーブルの表示スイッチを操作してもよい。

コマンド 85H

パラメータ (1 バイト)

**(13) SPROV コマンド**

スプライトオーバー/衝突の情報を獲得する。

コマンド 81H

パラメータ なし

受取りパラメータ 1 バイト

0	SO	CD	OVS
---	----	----	-----

SO スプライトオーバー (SPRON コマンドで設定した数より多くのスプライトが同一ラスタに存在すること) の発生を示すフラグ

CD 衝突が検出されたことを示すフラグ

OVS スプライトオーバーした最初のスプライト番号

[解説]

このコマンドは、VRTC 期間の最初の $160\mu\text{S}$ の間に発行する必要がある。

また、TSP ステータスの SC ビット (4.3.4 参照) はこのコマンドによりクリアされる。

4.5 グラフィック機能

4.5.1 グラフィック機能概要

VA のグラフィック機能には次の様な特徴がある。

- マルチプレーンモードと 6 万色表示のシングルプレーンモード
- 2 枚のグラフィック画面と独立した水平表示解像度設定
- フレームバッファによるスムーズスクロール
- 画面分割機能
- 高度な GVRAM 拡張アクセス機能
- 専用プロセッサによる BLT 機能 / LINE 機能

大きなモードとしてマルチプレーンモードとシングルプレーンモードとがある。マルチプレーンモードはカラーコードの各ビットごとに表示プレーンを持つ方式で、各プレーンにおいてはピクセルと GVRAM ビットとが一意に対応している。このモードは 88M/F シリーズや PC9801 シリーズと同じ方式である。シングルプレーンモードは 1 つのプレーンで多色の表示を行うもので、GVRAM の複数のビットで 1 つのピクセルを表す。このモードでは最大 65536 色の表示やビデオディジタイズ (オプション) が可能となっている。

シングルプレーンモードとマルチプレーンモードを切り換えるためには下記の 2 つのポートを設定する。どちらのポートも同じモードに設定して使用する。

ポート 100H (ワード) : 表示の制御

ビット 10 : 0 = マルチプレーンモード

1 = シングルプレーンモード

ポート 152H (ワード) : 描画の制御

ビット 12 : 0 = マルチプレーンモード

1 = シングルプレーンモード

● グラフィック表示解像度

グラフィック表示画面の表示解像度はテキスト画面やスプライト画面とは独立に設定することができる。またシングルプレーンモードでは 2 枚のグラフィック画面が使用できるが、各々に対しても水平解像度は独立に設定することができる。

表示解像度は使用する CRT のタイプ及びその表示モードにより以下のものが設定可能である。

●水平解像度

320／640 ドット

●垂直解像度

15.98KHz ノンインターレース : 200／204 ライン

15.73KHz インターレース : 400／408 ライン

24.8KHz ノンインターレース : 400／408 ライン

表示解像度及び CRT の表示モードの設定は以下のポートにより行う。

ポート 102H (ワード)

ビット 4 グラフィック画面 0 水平解像度

0 : 640 ドット

1 : 320 ドット

ビット 12 グラフィック画面 1 水平解像度

0 : 640 ドット

1 : 320 ドット

ポート 100H (ワード)

ビット 1 ビット 0 グラフィック画面垂直解像度(画面 0／1 共通)

0 0 : 400 ドット

0 1 : 408 ドット

1 0 : 200 ドット

1 1 : 204 ドット

4.5.2 マルチプレーンモード

マルチプレーンモードにおいてはグラフィック VRAM (GVRAM) は 4 枚のプレーン構造となっており、これによってカラー表示及びモノクロ表示が行われる。このモードのグラフィック表示画面は常に 1 つ (グラフィック画面 0) しかサポートされていない。

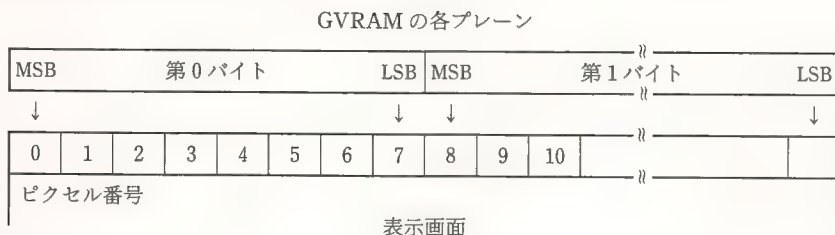
(1) GVRAM

GVRAM は最大 256K バイトの容量を持っている。CPU のメモリ空間にマッピングされる際には V3 対応モードと V1／V2 対応モードによって配置が異なる。V3 対応モードでは GVRAM は各プレーンあたり 64K バイトの容量を持っており、CPU のメモリ空間の A0000H ～DFFFFH 番地に 256K バイトのバンクとしてマップされる。

● GVRAM マップ (独立アクセスモード)

DFFFFH	プレーン 3
D0000H	プレーン 2
C0000H	プレーン 1
B0000H	プレーン 0
A0000H	

● GVRAM と表示画面との対応



(2) ピクセルサイズ

マルチプレーンモードの表示には3種類のピクセルサイズがあり、各々モノクロ表示、16色カラー表示、8色カラー表示に対応している。

- ① 1ビット／ピクセル (モノクロ表示)
- ② 4ビット／ピクセル (16色カラー表示)
- ③ 3ビット／ピクセル (8色カラー表示)

ピクセルサイズの設定はポート 102H, 110H により行われる。

ポート 102H

ビット1 ビット0 (グラフィック画面0 ピクセルサイズ)

0 0 : 1ビット／ピクセル

0 1 : 4 または 3ビット／ピクセル

ポート 110H

ビット7: プレーン3画面スイッチ

0 = 3ビット／ピクセル (プレーン3オフ)

1 = 4ビット／ピクセル (プレーン3オン)

(2.1) 1ビット／ピクセル (モノクロ表示)

各プレーンは独立な表示プレーンとして扱われ、それぞれがモノクロのピクセルデータを格納している。表示を行う際には4プレーンある GVRAM 中から任意に複数のプレーンを選択して合成することができる。OR 合成が行われたピクセルデータは、テキスト画面のキャラクタジェネレータが持つデータに合成されてテキスト画面として扱われ、テキスト画面のアトリビュートによって修飾されて表示される。

表示ピクセル = (プレーン0) OR (プレーン1) OR
(プレーン2) OR (プレーン3) OR (テキスト)

合成表示を行うプレーンの選択はポート 110H によって指定される。

ポート 110H

ビット3: プレーン3画面スイッチ (1=オン / 0=オフ)

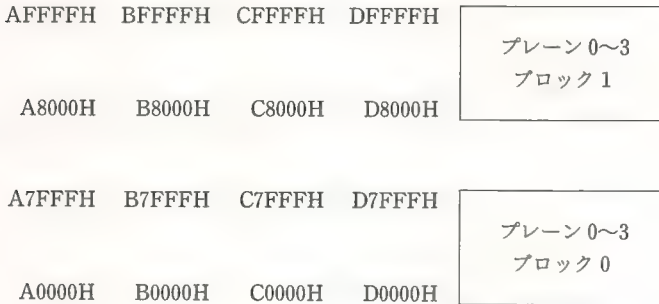
ビット2: プレーン2画面スイッチ (1=オン / 0=オフ)

ビット1: プレーン1画面スイッチ (1=オン / 0=オフ)

ビット0: プレーン0画面スイッチ (1=オン / 0=オフ)

(3.2) 拡張アクセスモード

指定された任意の複数のプレーンを同時にアクセスできる。アクセスするアドレスはどのプレーンでもよい。また、各プレーンは2つのブロックに分割され、各々のブロックは切り換えないとアクセスできない。ただしこの分割は表示機能とは無関係である。



(3.2.1) アクセスブロックの選択

アクセスブロックの選択は次のポートで行われる。

ポート 512H ビット 0
 0 : ブロック 0 選択
 1 : ブロック 1 選択

(3.2.2) プレーンの選択

同時にアクセスするプレーンの選択は次のポートを用いる。ここで選択されなかったプレーンはアクセス時に無視される。

ポート 514H : 読出しプレーン選択
 ポート 516H : 書込みプレーン選択

(3.2.3) 比較読出し機能

拡張アクセスモードでは比較読出し機能を使用できる。この機能の使用には、ポート 518H を用いる。

ポート 518H ビット 5
 0 : 比較読出し機能を使用しない
 1 : 比較読出し機能を使用する

比較読出し機能を使用しない場合のデータリードは、選択された各プレーンからのデータの AND をとったものが返される。

比較読出し機能使用時のデータリードシーケンスは次のようになる。

- ① 選択された各プレーンに対して
 GVRAM からデータを読み出す。
 プレーンごとの比較データレジスタと XNOR をとる。
- ② 各プレーンの結果をすべて AND する。
- ③ その結果を CPU に返す。

つまり、全プレーンについて比較データレジスタと一致したビットが1となった値が読み出される。

比較データレジスタは1バイトであり、ワードアクセスの時は上位バイト／下位バイトに同じデータが用いられる。比較データレジスタへの値の設定は次のようにして行う。

① 4つの比較データレジスタに 00H または FFH を設定する場合

ポート 528H

ビット 3：プレーン 3 用比較データレジスタ設定

ビット 2：プレーン 2 用比較データレジスタ設定

ビット 1：プレーン 1 用比較データレジスタ設定

ビット 0：プレーン 0 用比較データレジスタ設定

1 = FFH を設定

0 = 00H を設定

各ビットに設定したい値 (1/0) を決めて OUT する。

② 任意の値を設定する場合

ポート 520H：プレーン 0 用比較データレジスタ

ポート 522H：プレーン 1 用比較データレジスタ

ポート 523H：プレーン 2 用比較データレジスタ

ポート 524H：プレーン 3 用比較データレジスタ

各ポートに設定したい値 (バイト) を OUT する。

(3.2.4) 拡張ライト機能

拡張アクセスモードにおける GVRAM ライトモードは 4 種ある。

- | | |
|---------------|---------------------|
| ① NOP | 書込み動作をしない |
| ② CPU データライト | CPU からのデータをそのまま書き込む |
| ③ パターンレジスタライト | パターンレジスタの値を書き込む |
| ④ LU 出力ライト | 論理演算の結果を書き込む |

各モードの切換えにはポート 156H を使用する。

ビット 4 ビット 3

0	0	LU 出力ライト (リセット時)
0	1	パターンレジスタライト
1	0	CPU データライト
1	1	NOP

① NOP

CPU からの GVRAM ライト動作を無視する (書込み禁止)。

② CPU データライト

CPU からのデータを選択された全プレーンに書き込む。

③パターンレジスタライト

CPU からのデータは無視し、各プレーン毎のパターンレジスタの値をライトする。
パターンレジスタは2バイトからなり、上位バイトを使用するかどうかが指定できる。

ポート 518H ビット 2

0 = 8 ビットモード（上位バイトは使用しない）

1 = 16 ビットモード（上位バイトを使用する）

8 ビットモードでワードアクセスを行った場合は下位バイトと同じものが上位バイトにも使用される。

16 ビットモードでバイトアクセスを行った場合は下位バイト／上位バイトが交互に使用される。次に使用されるのがどちらのバイトであるかの読出し／設定はポート 550H で行うことができる。

ポート 550H (IN／OUT)

ビット 3：プレーン 3 パターンレジスタ使用開始バイト

ビット 2：プレーン 2 パターンレジスタ使用開始バイト

ビット 1：プレーン 1 パターンレジスタ使用開始バイト

ビット 0：プレーン 0 パターンレジスタ使用開始バイト

0 = 下位バイトから使用開始

1 = 上位バイトから使用開始

上位バイトから使用開始に設定すると、ワードアクセスにおいて上位バイト／下位バイトが逆転するので注意。

パターンレジスタへ値を設定する際のモードには4種類あり、ポート 518H で指定する。

ビット 1 ビット 0

0	0	固定モード
0	1	メモリリード時に更新
1	0	メモリライト時に更新
1	1	メモリリード／ライト時に更新

固定モード：ポートから設定された値を常に使用する。

〈パターンレジスタの設定ポートアドレス〉

	下位バイト	上位バイト
プレーン 0	530H	540H
プレーン 1	532H	542H
プレーン 2	534H	544H
プレーン 3	536H	536H

メモリリード時更新

CPU のメモリリード時のデータでパターンレジスタを更新する。

メモリライト時更新

CPU のメモリライト時のデータでパターンレジスタを更新する。

メモリリード／ライト時更新

リード／ライトどちらのデータでもパターンレジスタを更新する。

メモリアクセスによるパターンレジスタの更新も、バイトアクセス時に次に上位バイトから行うか下位バイトから行うかを指定できる。

ポート 552H (IN/OUT)

ビット 3：プレーン 3 パターンレジスタ更新開始バイト

ビット 2：プレーン 2 パターンレジスタ更新開始バイト

ビット 1：プレーン 1 パターンレジスタ更新開始バイト

ビット 0：プレーン 0 パターンレジスタ更新開始バイト

0 = 下位バイトから更新開始

1 = 上位バイトから更新開始

④ LU 出力ライト

CPU からのデータ／書き込み先データ／パターンレジスタの 3 種のデータ間の論理演算の結果を書き込む。

論理演算の方法は、ROP (ラストオペレーション) コードと呼ばれる 1 バイトの値で指定する。ROP コードは次のようにして求める。

P = パターンレジスタのデータ

S = CPU データ

D = 書き込み先データ (GVRAM データ)

とした時にこの P S D の組合せには 8 通りあり、各組合せに対する出力は計 8 ビットで表される。この出力パターンを ROP コードとして ROP レジスタに設定する。

P	S	D	出力データパターン					
0	0	0	0	1	0		0	1
0	0	1	0	0	1		1	1
0	1	0	0	0	0		1	1
0	1	1	0	0	0	1	1
1	0	0	0	0	0		1	1
1	0	1	0	0	0		1	1
1	1	0	0	0	0		1	1
1	1	1	0	0	0		1	1
			00	01	02	FEH	FFH

LSB
 ↑
 ↓
 MSB
 ROP コード

ROP レジスタはプレーン毎に用意されており、下記のポートで設定できる。

ポート 560H：プレーン 0 用 ROP レジスタ

ポート 562H：プレーン 1 用 ROP レジスタ

ポート 564H：プレーン 2 用 ROP レジスタ

ポート 566H：プレーン 3 用 ROP レジスタ

4.5.3 シングルプレーンモード

シングルプレーンモードにおいてはグラフィック表示画面は2枚の独立したグラフィック画面(グラフィック画面0, グラフィック画面1)によって構成される。2枚のグラフィック画面にはそれぞれ独立に水平解像度と表示ピクセルサイズを設定できる。またこのモードではSGP(4.5.6参照)が使用できる。

(1) GVRAM

シングルプレーンモードでのグラフィック画面にはグラフィック画面0/1の2つの表示画面を定義することができる。その時、256K バイトの GVRAM は以下の2種のモードに従って各グラフィック画面に割り付けられる。

ポート 100H(ワード) ビット 11

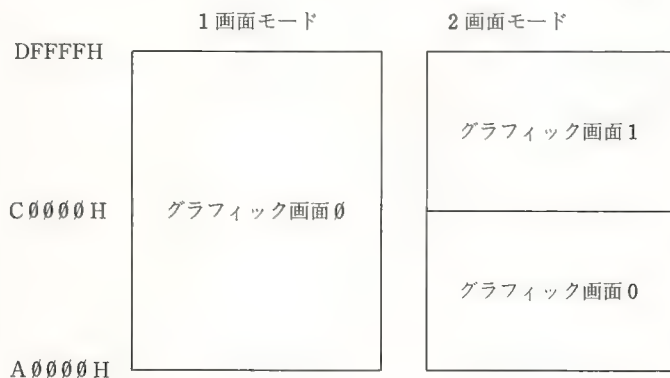
0:1 画面モード(グラフィック画面0 しか使用できない)

1:2 画面モード(グラフィック画面0/1 とともに使用可)

1 画面モードが指定されると GVRAM はすべてグラフィック画面0 にマッピングされて表示が行われる。このモードではグラフィック画面1 は使用することができない。

2 画面モードが指定されると GVRAM は、2つの 128K バイトブロックに分割され、各々がグラフィック画面0/グラフィック画面1 に対応する。なお、グラフィック画面0 に16ビット/ピクセル(後述)が設定された場合は、グラフィック画面1 は使用できない。

● GVRAM マップ



GVRAM と表示画面との対応は、次のピクセルサイズの項で説明する。

(2) ピクセルサイズ

シングルプレーンモードでの表示におけるピクセルサイズには4種類あり、各々表示可能な色数が異なる。

- ① 1ビット/ピクセル (4096 色中モノクロ表示)
- ② 4ビット/ピクセル (4096 色中16色表示)
- ③ 8ビット/ピクセル (256 色表示)
- ④ 16ビット/ピクセル (65536 色表示)

ピクセルサイズの設定はポート 102H(ワード)により行われる。

ビット 1 ビット 0 (グラフィック画面 0 ピクセルサイズ)

0	0	: 1ビット/ピクセル
0	1	: 4ビット/ピクセル
1	0	: 8ビット/ピクセル
1	1	: 16ビット/ピクセル

ビット 9 ビット 8 (グラフィック画面 1 ピクセルサイズ)

0	0	: 1ビット/ピクセル
0	1	: 4ビット/ピクセル
1	0	: 8ビット/ピクセル
1	1	: グラフィック画面 0 に 16 ビット/ピクセルを設定したとき

(2.1) 1ビット/ピクセル

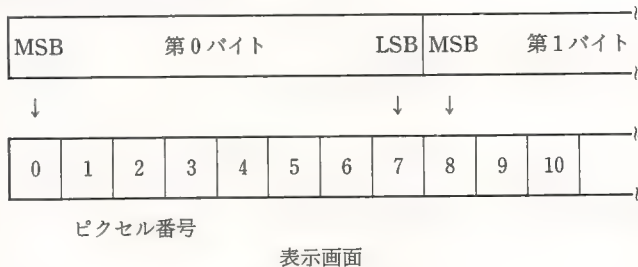
このモードでは 16 個のカラーパレットからフォアグラウンドカラーを 1 色指定できる。バックグラウンドカラーは常にカラーコード 0 である。

フォアグラウンドカラーはポート 110H で設定する。

ポート 110H(ワード)

ビット 11～8 : フォアグラウンドカラーコード
(パレット番号)の設定

● GVRAM と表示画面との対応

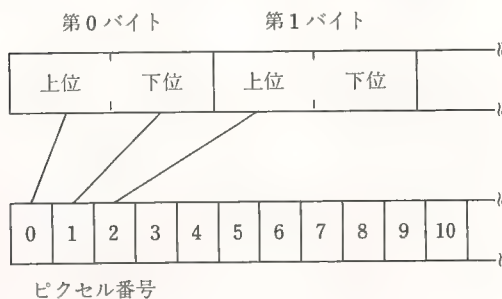


(2.2) 4ビット/ピクセル

このモードではピクセルごとにカラーパレットを指定することにより、4096 色中 16 色を同時に表示できる。

● GVRAM と表示画面の対応

各ニブルに 0～15 のカラーコードを設定する。



(2.3) 8ビット／ピクセル

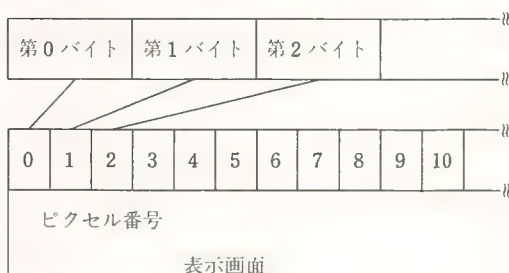
このモードには、カラーパレットを使用するパレット指定画面モードとカラーパレットを使用しない直接色指定画面モードとがある。

パレット指定モードでは各ピクセルごとに32個のカラーパレットを指定することにより、4096色中32色の同時表示が可能である。この場合、8ビットあるピクセルデータの下位5ビットがカラーパレットを指定するのに使われる。その場合はカラーパレットモード3(後述)を使用する。

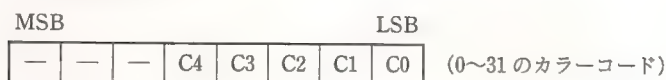
直接色指定モードでは各ピクセル単位に1色／256色を指定する。カラーパレットが使用されないため、表示色は8ビットのデータによりダイレクトにRGB指定される。

パレット指定モードと直接色指定モードとの切換えは、8ビット／ピクセルのグラフィック画面を優先表示画面0～3に割り当てる(パレット指定モード)か、優先表示画面4/5(直接色指定モード)に割り当てるかによって決定される。

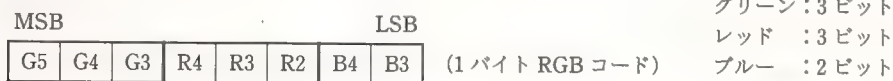
● GVRAM と表示画面との対応



・パレット指定モード



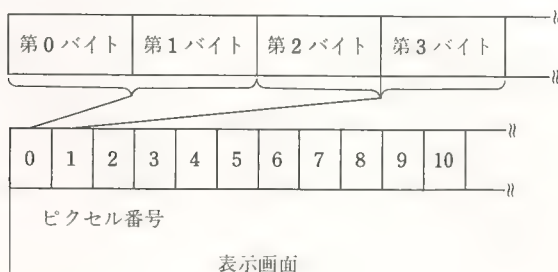
・直接色指定モード



(2.4) 16ビット／ピクセル

このモードでは各ピクセル単位に1色／65536色を指定する。カラーパレットが使用されないため、表示色は16ビットのデータよりダイレクトにRGB指定される。またこのモードはグラフィック画面0にのみ設定でき、その場合グラフィック画面1は使用できない。

● GVRAM と表示画面との対応



・ワード内ビットアサイン

MSB			上位バイト			LSB MSB			下位バイト			LSB			
G5	G4	G3	G2	G1	G0	R4	R3	R2	R1	R0	B4	B3	B2	B1	B0

グリーン：6ビット
 レッド：5ビット
 ブルー：5ビット

(3) 拡張アクセス機能

シングルプレーンモードでは拡張アクセスと独立アクセスとの区別はなく、修飾のない GVRAM アクセスをしたい場合には下記の拡張ライト機能の② CPU データライトに設定する。

(3.1) 拡張リード機能

シングルプレーンモードには比較読出し機能はない。

(3.2) 拡張ライト機能

GVRAM へのライトモードはマルチプレーンモードと同様に 4 種類ある。

- ① NOP 書込み動作をしない
- ② CPU データライト CPU からのデータをそのまま書き込む
- ③ パターンレジスタライト パターンレジスタの値を書き込む
- ④ LU 出力ライト 論理演算の結果を書き込む

各モードの切換えはポート 580H を使用する。これはグラフィック画面 0/1 に共通である。

ポート 580H

ビット 4	ビット 3	
0	0	LU 出力ライト(リセット時)
0	1	パターンレジスタライト
1	0	CPU データライト
1	1	NOP

① NOP

CPU からの GVRAM ライト動作を無視する(書込み禁止)。

② CPU データライト

CPU からのデータをそのまま書き込む(無修飾ライト)。

③パターンレジスタライト

CPU からのデータは無視し、パターンレジスタの値を書き込む。

パターンレジスタはつねに 2 バイトであり、メモリアクセス時の自動更新モードはない。

パターンレジスタのポートは次のようになっており、書き込むアドレスによって使用されるパターンレジスタが異なる。

ポート 590H：A0000-BFFFFH 用パターンレジスタ(ワード)

ポート 592H：C0000-DFFFFH 用パターンレジスタ(ワード)

④ LU 出力ライト

CPU からのデータ／書き込み先データ／パターンレジスタの3種のデータ間の論理演算の結果を書き込む。

論理演算の方法は、ROP(ラスタオペレーション)コードと呼ばれる1バイト値で指定する。ROP コードはマルチプレーンモードと同じである。

ROP レジスタのポートは次のようになっており、書き込むアドレスによって使用されるROP レジスタが異なる。

ポート 5A0H：A0000-BFFFFH 用 ROP レジスタ(バイト)

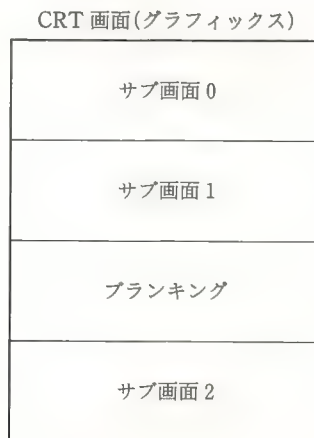
ポート 5A2H：C0000-DFFFFH 用 ROP レジスタ(バイト)

4.5.5 フレームバッファと画面分割・スクロール

(1) 画面分割機能(サブ画面)

グラフィック画面は最大4個のサブ画面を定義することができ、これにより画面を最大4分割することが可能である。サブ画面は水平方向にのみ分割された画面で、その表示位置と高さはライン単位に設定することができる。

●画面分割の例



4つのサブ画面はそれぞれグラフィック画面0／グラフィック画面1に割り当てられている。

サブ画面0：グラフィック画面0用

サブ画面1：グラフィック画面1用

サブ画面2：グラフィック画面0用

サブ画面3：グラフィック画面0用

グラフィック画面0には3つのサブ画面が割り当てられているが、CRT画面の上から順にサブ画面0, 2, 3と使用する必要がある。またサブ画面間に間が空いた設定をすることは可能だが、同じグラフィック画面に割り当てられたサブ画面(0, 2, 3)は互いにオーバーラップしないように設定しなくてはならない。

マルチプレーンモードではグラフィック画面0しか使用できないため、サブ画面1は使用できず、画面は最大3分割までとなる。

画面分割を行わない場合は、グラフィック画面 0 では通常サブ画面 0 を、グラフィック画面 1 ではサブ画面 1 を画面全体に割り当てて使用することになる。

(2) フレームバッファ

各サブ画面はそれぞれ仮想フレームバッファを持っている。フレームバッファは GVRAM 中にあり、そのスタートアドレス・横幅・縦幅によって設定される。フレームバッファ中の指定した一部（実表示画面）がサブ画面として表示される。

サブ画面とフレームバッファとは一意に対応している。

フレームバッファ 0：サブ画面 0 グラフィック画面 0

フレームバッファ 1：サブ画面 1 グラフィック画面 1

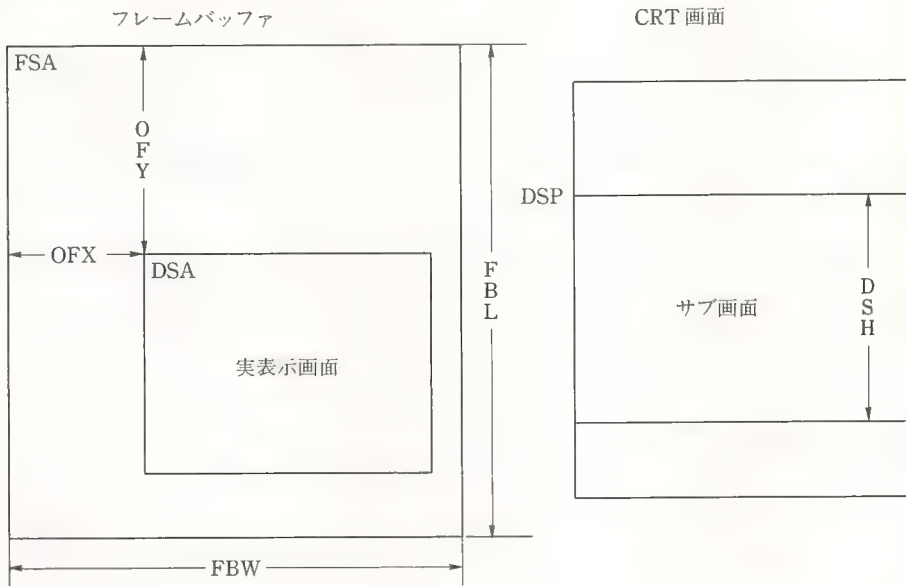
フレームバッファ 2：サブ画面 2 グラフィック画面 0

フレームバッファ 3：サブ画面 3 グラフィック画面 0

フレームバッファは互いにオーバーラップした設定を行ってもよい。またフレームバッファ 1 は機能に一部制限があり、ラップアラウンド機能(後述)が使用できない。

(3) フレームバッファとサブ画面

フレームバッファ内の任意のアドレスを指定して実表示画面を設定する。実表示画面の幅は常に CRT 画面の幅と一致している。高さはサブ画面の高さによって指定される。



フレームバッファおよびサブ画面の制御パラメータとしては次のものがある。スタートアドレスおよび横幅は 4 バイトバウンダリに設定しなければならない。

FSA	：フレームバッファスタートアドレス	OFY	：実表示画面 Y オフセット
FBW	：フレームバッファ横幅	DSA	：実表示画面スタートアドレス
FBL	：フレームバッファ縦幅	DSP	：サブ画面表示位置
OFX	：実表示画面 X オフセット	DSH	：サブ画面高さ

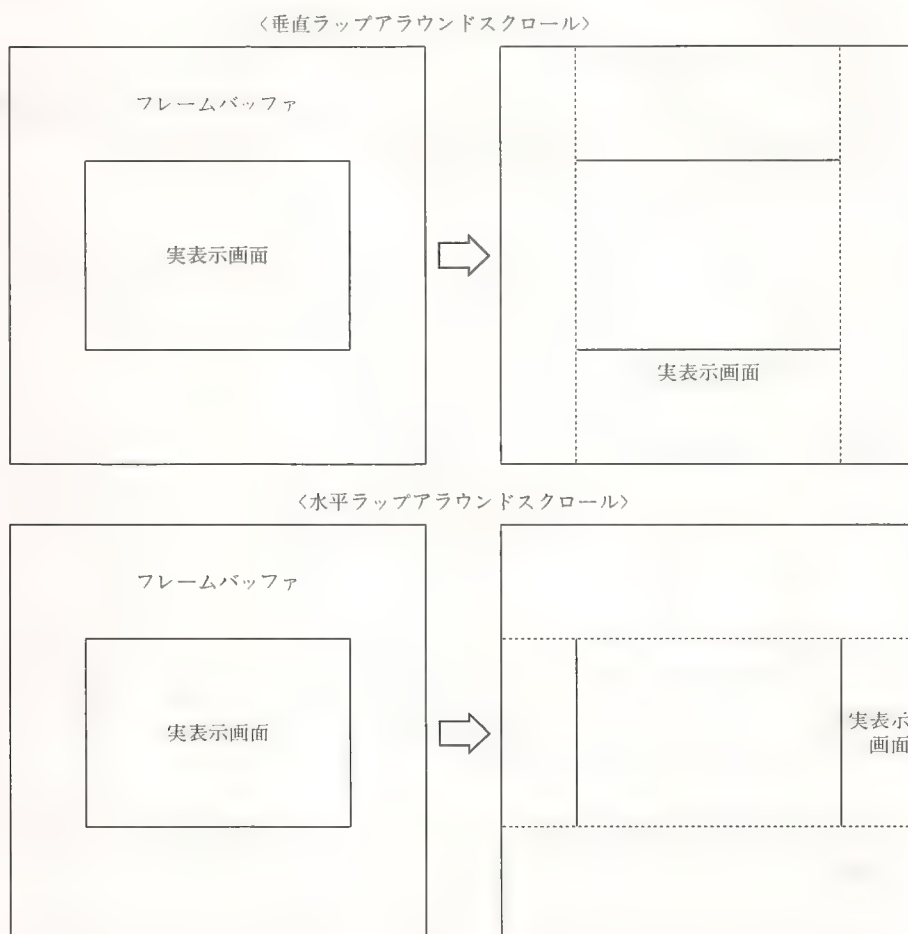
(4) スクロール

表示画面のスクロールはフレームバッファ中の実表示画面の位置を変えることにより、サブ画面単位に垂直／水平方向でドット単位に行うことができる。このとき変更するグラフィック画面制御パラメータは次の3つである。

- 実表示画面の X オフセット
- 実表示画面の Y オフセット
- 実表示画面の表示開始アドレス

またフレームバッファのサイズを超えるスクロールを行う場合、グラフィック画面0に属する実表示画面はフレームバッファの大きさを考慮して水平／垂直方向ともにラップアラウンドされる。ただしモードによっては制限が加わることがある(後述)。

グラフィック画面1(フレームバッファ1)では水平／垂直ともにラップアラウンドはサポートされておらず、実表示画面がフレームバッファをはみだす設定を行ってはならない。



●水平ラップアラウンド時の制限

- ①マルチプレーンモードの1ビット／ピクセル
水平ラップアラウンドはサポートされない。
- ②シングルプレーンモードでグラフィックスを2画面表示したとき
グラフィック画面0で水平ラップアラウンドを行うと、グラフィック画面1の表示画面設定に対して制限が加わる(グラフィック画面0の設定には制限はない)。

水平解像度	1ビット／ピクセル	4ビット／ピクセル	8ビット／ピクセル
640ドット	○	△	×
320ドット	○	○	△

○：制限なし
×：使用不可
△：表示スタートアドレスとフレームバッファの横幅が128バイト単位でしか設定できない

(5) グラフィック画面制御パラメータ

フレームバッファ／サブ画面の各種パラメータはI/Oポートに割り当ててあるグラフィック画面制御パラメータで設定する。グラフィック画面制御パラメータのポート位置は次のようになっている。

- フレームバッファ0：ポート 200～217H
- フレームバッファ1：ポート 220～237H
- フレームバッファ2：ポート 240～257H
- フレームバッファ3：ポート 260～277H

以下に1つのフレームバッファのグラフィック画面制御パラメータの構造を示す。

(5.1) フレームバッファのスタートアドレス(FSA)

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
+00H	FSA														0	0
+02H	0	0	0	0	0	0	0	0	0	0	0	0	0	0	FSA	

フレームバッファのスタートアドレスを18ビットのバイトアドレスで設定する。ただし設定単位は4バイトバウンダリとなっているため、下位2ビットが常に[0]となっている必要がある。スタートアドレスには、CPUアドレスーA0000Hを設定する。

(5.2) フレームバッファの横幅(FBW)

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
+04H	0	0	0	0	0	FBW										0	0

フレームバッファの横幅を11ビットのバイト数で設定する。ただし設定単位は4バイトバウンダリとなっているため、下位2ビットは常に [0] となっている必要がある。

FBW の値	:	フレームバッファの横幅(バイト)
0	:	0
4	:	4
⋮	:	⋮
2040	:	2040
2044	:	2044

(5 . 3) フレームバッファの縦幅(FBL)

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
+06H	0	0	0	0	0	0	FBL									

フレームバッファの縦幅を 10 ビットのライン数で設定する。フレームバッファ 1 には設定できない。縦幅には実際の値-1 を設定する。

FBL の値	:	フレームバッファの縦幅(ライン数)
0	:	1
1	:	2
2	:	3
⋮	:	⋮
1022	:	1023
1023	:	1024

(5 . 4) 実表示画面の X オフセット (OFX) とドットアドレス

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
+08H	0	0	0	0	0	0	0	0	0	0	0	ドットアドレス				
+0AH	0	0	0	0	0	OFX										0 0

OFX は、フレームバッファ内の実表示画面の位置を示す X オフセットを 11 ビットのバイト数で設定する。ただし設定単位は 4 バイトバウンダリとなっているため、下位 2 ビットは常に [0] となっている必要がある。

OFX はフレームバッファ 1 には設定できない。

OFX の値	:	実表示画面の X オフセット(バイト数)
0	:	0
4	:	4
⋮	:	⋮
2040	:	2040
2044	:	2044

OFX では 4 バイトバウンダリでしか指定できないため、指定した 4 バイトの中の何ピクセル目から表示するかを、5 ビットのドットオフセットとして指定する。ドットオフセットはマルチプレーンモードの 1 ビット／ピクセルでは使用できない。この場合はテキスト／スプライトプロセッサに水平スミーズスクロールオフセットとして設定する。

各ピクセルサイズごとのドットオフセットの設定は次のとおり。

		表示モード別の表示開始ピクセル			
		シングルプレーンモード			
		[1]	[4]	[8]	[16]
ドットアドレスの値					
0	:	0	0	0	0
1	:	1	1	1	—
2	:	2	2	—	—
3	:	3	3	—	—
4	:	4	—	—	—
5	:	5	—	—	—
⋮	:	⋮			
15	:	15	—	—	—
16	:	16	4	2	1
17	:	17	5	3	—
18	:	18	6	—	—

		マルチプレーンモード	
		[4]	／ピクセル
		0	ピクセル目から表示
0	:	0	
1	:	1	
2	:	2	
3	:	3	
4	:	4	
5	:	5	
⋮	:		
15	:	—	
16	:	—	
17	:	—	
18	:	—	

19	:	19	7	—	—	—
20	:	20	—	—	—	—
⋮	:	⋮				
31	:	31	—	—	—	—

(5.5) 実表示画面のY オフセット (OFY)

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
+0CH	0	0	0	0	0	0	OFY									

フレームバッファ内の実表示画面の位置を示す Y オフセットを 10 ビットのライン数で設定する。

OFY の値	:	実表示画面の Y オフセット(ライン数)
0	:	0
1	:	1
2	:	2
⋮	:	⋮
1022	:	1022
1023	:	1023

(5.6) 実表示画面の表示開始アドレス (DSA)

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
+0EH	DSA															0	0
+10H	0	0	0	0	0	0	0	0	0	0	0	0	0	0	DSA		

フレームバッファ内の実表示画面の位置を 18 ビットのバイトアドレスで設定する。ただし設定単位は 4 バイトバウンダリとなっているため、下位 2 ビットは常に [0] となっている必要がある。OFX・OFY から計算されるアドレスとこの DSA とは同じになるように設定する。表示開始アドレスには、CPU アドレス—A0000H を設定する。

(5.7) サブ画面の高さ (DSH)、表示位置 (DSP)

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
+12H	0	0	0	0	0	0	0	DSH								
+16H	0	0	0	0	0	0	0	DSP								

(注：+14H のポートはない)

DSH は、サブ画面(実表示画面)の高さを 9 ビットのライン数で設定する。

DSH の値	:	サブ画面の高さ(ライン数)
0	:	0
1	:	1
2	:	2
⋮	:	⋮
510	:	510
511	:	511

DSP は、サブ画面の表示位置を 9 ビットのライン位置で設定する。

DSP の値	:	サブ画面の表示位置(ライン位置)
0	:	0
1	:	1
2	:	2
⋮	:	⋮
510	:	510
511	:	511

4.5.6 SGP(スーパーグラフィックプロセッサ)

SGP はグラフィック描画を高速で行うための専用プロセッサであり、RAM/漢字 ROM の長方形 BLT(ブロックトランスファ)機能、LINE 描画機能、ペイント補助機能などがある。SGP はシングルプレーンモードでのみ使用することができる。

(1) SGP のメモリマップ

SGP への命令で与えるアドレスは CPU 空間のアドレスではなく、すべて SGP 空間のアドレスを使用する。SGP は 4M バイトのメモリ空間を持ち、メイン ROM と辞書 ROM 以外のメモリをアクセスすることができる。そのメモリレイアウトは次のようになっている。メイン RAM は SGP アドレスと CPU アドレスが一致している。

VDP アドレス		メイン CPU アドレス
240000～3FFFFFFH	未使用	
200000～23FFFFFFH	GVRAM	A0000H～DFFFFFFH
190000～1FFFFFFH	未使用	
180000～18FFFFFFH	TVRAM	A0000H～AFFFFFFH
140000～17FFFFFFH	漢字 ROM #2	A0000H～DFFFFFFH
100000～13FFFFFFH	漢字 ROM #1	A0000H～DFFFFFFH
0A0000～0FFFFFFH	拡張メモリ	
040000～09FFFFFFH	拡張メイン RAM	40000H～9FFFFFFH
000000～03FFFFFFH	標準メイン RAM	00000H～3FFFFFFH

(2) SGP の制御

(2.1) SGP への命令設定法

SGP はメモリ上に展開された命令テーブルを順次実行していく。命令を置くメモリは通常メイン RAM が使用される。また SGP はワークメモリとして指定した 58 バイトの RAM を使用する。実行開始アドレスはポート 500～503H で設定する。このとき使用されるアドレスも SGP 空間アドレスである。指定するアドレスは偶数アドレスでなければならない。

ポート 500H：SGP 実行開始アドレス下位 16 ビット

ポート 502H：SGP 実行開始アドレス上位 16 ビット

実行開始アドレスを設定した後、実行開始指令を出す。

OUT 506H, 1

これにより、描画終了コマンドを実行するまで SGP は命令を実行し続ける。

(2.2) SGP の強制停止方法

通常は命令テーブルの最後に描画終了コマンドを置いて実行停止させるが、強制的に SGP を停止させることができる。

OUT 504H, 2

(2.3) SGP ステータス

SGP が命令を実行中かアイドル状態であるかのステータスを知ることができる。

IN 506H ビット 0

0 =SGP 停止

1 =SGP 動作中

(2.4) SGP 割込みの設定

SGP は描画終了コマンドを実行した段階でメイン CPU に割込みを発生することができる。割込み機能を使用するかどうかは次のポートで設定する。

OUT 504H ビット 4

0 =SGP 割込み禁止／割込みクリア

1 =SGP 割込み許可

SGP 割込みは INT 10H(割込みレベル 8)に接続されている。また、割込み要求は本ポートによりクリアする必要がある。

(3) SGP 命令

SGP のコマンド／パラメータは各々ワード単位で構成されている。命令の 1 ワード目はコマンドとして認識され、その後に必要なだけパラメータワードが続く。

(3.1) SGP コマンド

コマンドとして以下の 13 個が用意されている。

コマンド#	コマンド名	
01	END	描画終了
02	NOP	ノーオペレーション
03	SET WORK	作業領域設定
04	SET SOURCE	転送元ブロック指定
05	SET DESTINATION	転送先ブロック指定
06	SET COLOR	色指定
07	BITBLT	ブロックの転送
08	PTNBLT	繰り返しパターン転送
09	LINE	直線描画
0A	CLS	画面消去
0B	SCAN_RIGHT	ペイント補助機能
0C	SCAN_LEFT	ペイント補助機能

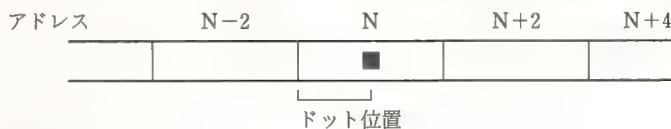
(3.2) SGP パラメータ

命令の 2 ワード目以降はパラメータである。以下にパラメータ指定時の概念を示す。

●ピクセル位置の指定

SGP パラメータにおいてピクセルの位置を指定するには、ピクセルが含まれるワードのアドレスと、当該ワード内でのドット位置により指定する。アドレスは SGP 空間内の物理アドレスであり、偶数バイトアドレスを指定する。ドット位置は、ワード内で左(MSB)から何番目のピクセルであるかを指定する。ピクセルサイズによってドット位置がとれる範囲は次の範囲に限られる。

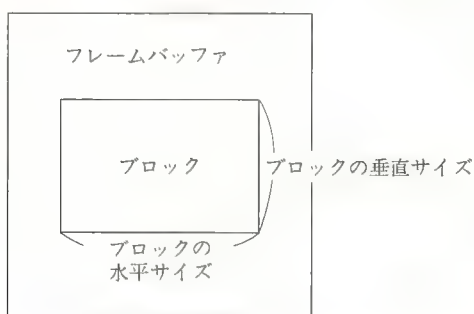
1ビット／ピクセル	0～15	8ビット／ピクセル	0または1
4ビット／ピクセル	0～3	16ビット／ピクセル	0のみ



●ブロック

BLTでは転送元・転送先を指定するのにブロック(メモリ中の仮想的長方形)を使用する。またLINEではブロックの対角線に直線を描画すると考える。ブロックの指定には、ブロックの開始ピクセル位置、ブロックの水平／垂直サイズ、フレームバッファの水平サイズ、ピクセルサイズが使用される。

ブロックの水平／垂直サイズにはピクセル数が、フレームバッファの水平サイズにはバイト数（したがって下位 2 ビットは 0）が使用される。



開始ピクセル位置がブロックのどの部分を指すかは BLT 時の転送方向、LINE の描画方向を考慮して決めなければならない。

- | | |
|------------------|-------------|
| ①上→下, 左→右に転送(描画) | ブロックの左上隅を指定 |
| ②上→下, 右→左に転送(描画) | ブロックの右上隅を指定 |
| ③下→上, 左→右に転送(描画) | ブロックの左下隅を指定 |
| ④下→上, 右→左に転送(描画) | ブロックの右下隅を指定 |

ピクセルサイズの指定には次の値(2 ビット)が使用される。

0	1ビット／ピクセル
1	4ビット／ピクセル
2	8ビット／ピクセル
3	16ビット／ピクセル

(4) 各命令の解説

(4.1) END

命令実行(コマンドフェッチ)を終了し、停止状態にはいる。パラメータはない。

[illegible]

(4.2) NOP

NOP 命令。パラメータはない。

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0

(4.3) SET WORK

SGP の作業領域を設定する。描画命令を使用する前に必ずこのコマンドで作業領域を設定する必要がある。作業領域として指定したアドレスから 58 バイトを使用する。作業領域は通常メイン RAM(00000-9FFFFH)に設定する。

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1

SGP ワークアドレス 下位 16 ビット	0
-----------------------	---

SGP ワークアドレス 上位 16 ビット

(4.4) SET SOURCE

ソースブロックを設定する。これは BITBLT・PATBLT の転送元ブロックとなる。

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0

0	0	0	0	0	0	0	0	0	開始ドット位置	0	0	SCRN_M
---	---	---	---	---	---	---	---	---	---------	---	---	--------

0	0	0	0	ソースブロックの水平サイズ(ピクセル単位)
---	---	---	---	-----------------------

0	0	0	0	ソースブロックの垂直サイズ(ピクセル単位)
---	---	---	---	-----------------------

ソースフレームバッファの横幅(バイトサイズ)	0	0
------------------------	---	---

ソースブロック開始アドレス(下位 16 ビット)	0
--------------------------	---

ソースブロック開始アドレス(上位 16 ビット)

SCRN_M : ピクセルサイズ
 0 1 ビット／ピクセル
 1 4 ビット／ピクセル
 2 8 ビット／ピクセル
 3 16 ビット／ピクセル

(4.5) SET DESTINATION

デスティネーションブロックを設定する。このコマンドは次の場合に使用される。

- ① BITBLT, PATBLT の転送先ブロックの設定
- ② SCAN コマンドの開始ピクセル・ピクセル数の設定

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1
0	0	0	0	0	0	0	0	開始ドット位置				0	0	SCRN_M	
0	0	0	0	デスティネーションブロックの水平サイズ											
0	0	0	0	デスティネーションブロックの垂直サイズ											
デスティネーションフレームバッファの横幅(バイトサイズ)														0	0
デスティネーションブロック開始アドレス(下位 16 ビット)															0
デスティネーションブロック開始アドレス(上位 16 ビット)															

SCRN_M	:	ピクセルサイズ
0		1 ビット／ピクセル
1		4 ビット／ピクセル
2		8 ビット／ピクセル
3		16 ビット／ピクセル

(4.6) SET COLOR

描画色を指定する。指定した色は次の場合に使用される。

- ① BITBLT, PATBLT で転送元が 1 ビット／ピクセルのとき
- ② LINE での直線描画時
- ③ CLS での画面消去時
- ④ SCAN で探す色の指定

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0
カラーコード／RGB コード															

①②④の場合、対象となるブロック／領域のピクセルサイズに合わせて次のビットが使用される。

1 ビット／ピクセルのとき	1 ビット×16ピクセル分
4 ビット／ピクセルのとき	4 ビット× 4 ピクセル分
8 ビット／ピクセルのとき	8 ビット× 2 ピクセル分
16ビット／ピクセルのとき	16ビット全部

(4.7) BITBLT

ソースブロックからデスティネーションブロックへブロック転送する。ソースブロック、デスティネーションブロックはそれぞれ SET SOURCE コマンド、SET DESTINATION コマンドで指定する。

[illegible]

0	0	0	SF	VD	HD	TP-MOD	0	0	0	0	LOGICAL-OP
---	---	---	----	----	----	--------	---	---	---	---	------------

SF 0：ソースをデスティネーションのビット位置に合わせてシフトして転送する
 1：ソースをシフトせずに転送する

VD 垂直転送方向
0：ブロックの上から下にアドレスを進めながら転送する
1：ブロックの下から上にアドレスを進めながら転送する

HD	水平転送方向
	0 : ブロックの左から右にアドレスを進めながら転送する
	1 : ブロックの右から左にアドレスを進めながら転送する

TP-MOD トランスペアレント転送モード

0：ソースをそのままデスティネーションに転送する

1：ソースブロックが0の部分は転送しない

2：デスティネーションブロックが0の部分だけ転送する

LOGICAL-OP		論理演算指定
0 : 0		8 : NOT(S OR D)
1 : S AND D		9 : NOT(S XOR D)
2 : NOT(S) AND D		A : NOT(S)
3 : NOP		B : NOT(S) OR D
4 : S AND NOT(D)		C : NOT(D)
5 : S		D : S OR NOT(D)
6 : S XOR D		E : NOT(S AND D)
7 : S OR D		F : 1

(4.8) PATBLT

ソースブロックからデスティネーションブロックへパターン転送する。BITBLT と同様の動作をするが、デスティネーションブロックがソースブロックより大きい場合にソースブロックが2次元的に繰り返されて転送される。

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0

0	0	0	SF	VD	HD	TP-MOD	0	0	0	0	LOGICAL-OP
---	---	---	----	----	----	--------	---	---	---	---	------------

※パラメータの内容は BITBLT と同じ。

(4.9) LINE

SET COLOR コマンドで設定した直線を描画する。

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1

0	0	0	0	VD	HD	TP-MOD	0	0	0	0	LOGICAL-OP
---	---	---	---	----	----	--------	---	---	---	---	------------

0	0	0	0	0	0	0	0	開始ドット位置	0	0	SCRN_M
---	---	---	---	---	---	---	---	---------	---	---	--------

LINE 水平サイズ(ピクセル単位)

LINE 垂直サイズ(ピクセル単位)

描画するフレームバッファの横幅(バイトサイズ)	0	0
-------------------------	---	---

描画開始アドレス(下位 16 ビット)	0
---------------------	---

描画開始アドレス(上位 16 ビット)

- VD 垂直描画方向
- 0：上から下への描画
 - 1：下から上への描画

- HD 水平描画方向
- 0：左から右への描画
 - 1：右から左への描画

* TP-MOD, LOGICAL-OP は BITBLT と同様。
* 開始ドット位置以下は SET SOURCE と同様。

(4.10) CLS

メモリを SET COLOR コマンドで指定した値でクリアする。

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0

CLS 開始アドレス(下位 16 ビット)	0
-----------------------	---

CLS 開始アドレス(上位 16 ビット)

CLS 領域ワードサイズ(下位 16 ビット)

CLS 領域ワードサイズ(上位 16 ビット)

(4.11) SCAN RIGHT

ペイントの補助機能。右方向へ SET COLOR コマンドで指定された色の点が見つかるまでスキャンする。スキャン開始ピクセル、スキャンの結果ともデスティネーションブロックのパラメータを使用する。

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1

●入力

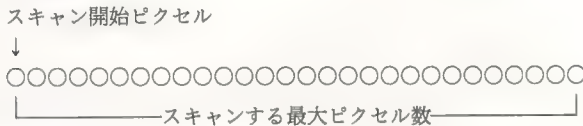
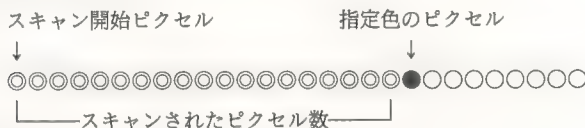
スキャン開始ピクセル： デスティネーションブロック開始ピクセル
 スキャン画面モード： デスティネーションの SCR_N_M
 スキャンする最大ピクセル数：デスティネーションブロックの水平サイズ
 スキャンする色： SET COLOR コマンドで指定した色

●出力

スキャンされたピクセル数： デスティネーションブロックの水平サイズ

指定した色のピクセルを発見した場合、スキャン開始ピクセルからのピクセル数をデスティネーションブロックの水平サイズに設定する。この直後に PATBLT コマンドを実行することにより、ペイント機能を実現することができる。

スキャン開始ピクセルが指定色の場合は、スキャンされたピクセル数は 0 になる。指定色が発見されなかった場合はデスティネーションブロックの水平サイズは更新されない。

●スキャン前**●スキャン後****(4.12) SCAN LEFT**

ペイントの補助機能。左方向へ SET COLOR コマンドで指定された色の点が見つかるまでスキャンする。スキャン開始ピクセル、スキャンの結果ともデスティネーションブロックのパラメータを使用する。

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1

●入力

スキャン開始ピクセル： デスティネーションブロック開始ピクセル
 スキャン画面モード： デスティネーションの SCR_N_M
 スキャンする最大ピクセル数：デスティネーションブロックの水平サイズ
 スキャンする色： SET COLOR コマンドで指定した色

●出力

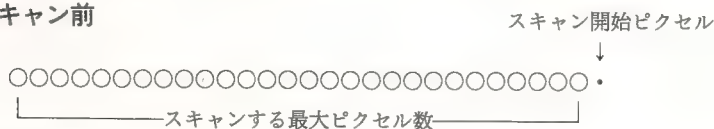
スキャンされた領域の左端： デスティネーション開始アドレス、ドット位置

スキャンされたピクセル数： デスティネーションブロックの水平サイズ

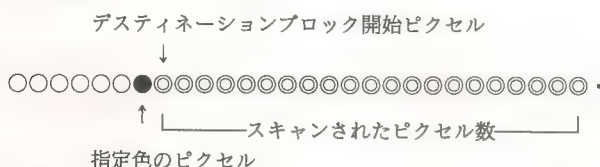
指定した色のピクセルを発見した場合、スキャン開始ピクセルからのピクセル数をデスティネーションブロックの水平サイズに設定する。この直後に PATBLT コマンドを実行することにより、ペイント機能を実現することができる。

スキャン開始ピクセルが指定色の場合、スキャンされたピクセル数は 0 になる。指定色が発見されなかった場合はデスティネーションブロックの水平サイズは更新されない。

●スキャン前



●スキャン後



(5) 使用上の制限

SGP 使用時には以下に述べるような制限がある。

- ① SGP はフレームバッファの境界判断を行わない。したがって、ラップアラウンドする BLT や LINE を行う場合は、フレームバッファの境界で区切って設定すること。
- ② BLT において転送元ブロックと転送先ブロックのピクセルサイズが異なる場合に転送可能なのは 1 ビット／ピクセルから多ビット／ピクセルだけである。
- ③ 1 ビット／ピクセルから多ビット／ピクセルへ BLT する場合、HD(水平転送方向)は必ず 0 (左→右)でなければならない。
- ④ HD(水平転送方向)が 1 (右→左)の場合、TP-MODE(トランスペアレントモード)は必ず 0 (ソースをそのまま転送)でなければならない。
- ⑤ ブロックの水平／垂直サイズ、フレームバッファの水平サイズを 0 に設定した場合の動作は保証しない。
- ⑥ 命令仕様で '0' と記されているビットは必ず 0 を指定すること。

(6) SGP 命令テーブルの例

(6.1) SGP 作業領域の設定

SET WORK コマンドを用いて SGP の作業領域を設定する。作業領域は描画する前に必ず設定を必要とするが、一度設定すれば以後設定する必要はない。

```
0003H ; SET WORK コマンド
0000H ; 作業領域を 10000H から設定。
0001H ;
0001H ; END コマンド
```

(6.2) グラフィック画面消去

4ビット／ピクセル，640×400ピクセルのフレームバッファ・表示画面をもつグラフィック画面を0で消去する。フレームバッファはCPUアドレスのA0000Hから始まるものとする。

0006H ; SET COLOR コマンド
 0000H ; 消去データ = 0
 000AH ; CLS コマンド
 0000H ; フレームバッファアドレスは
 0020H ; SGP 空間で 200000H から
 FA00H ; フレームバッファの大きさは
 0000H ; 64000 ワード
 0001H ; END コマンド

4.6 カラーとパレット

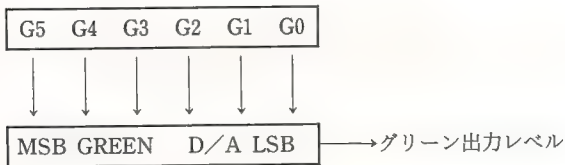
4.6.1 カラー階調

VA ではハードウェアとして RGB ごとに 6 ビットの D/A コンバータにより 64 段階の階調を持ち，その中から指定した階調の色が出力される。階調の指定のされかたを各ピクセルサイズごとに示す。

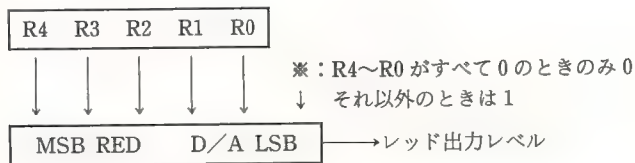
(1) 16 ビット／ピクセル (RGB 画面)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
G5	G4	G3	G2	G1	G0	R4	R3	R2	R1	R0	B4	B3	B2	B1	B0

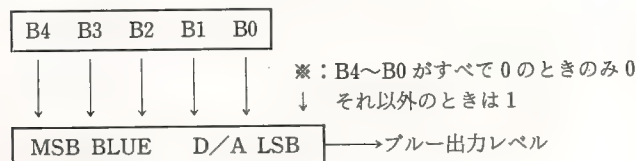
●グリーン：6ビット



●レッド：5ビット



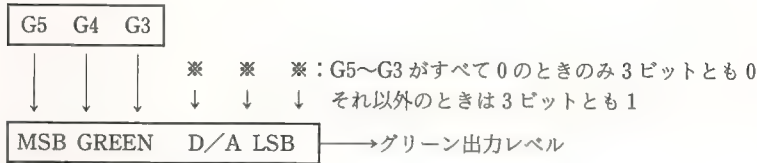
●ブルー：5ビット



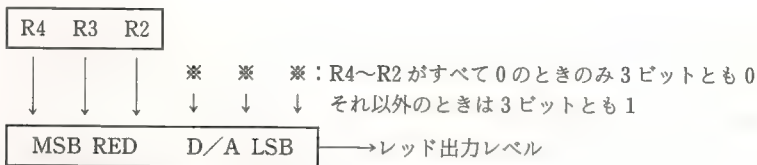
(2) 8ビット／ピクセル (RGB画面)

7	6	5	4	3	2	1	0
G5	G4	G3	R4	R3	R2	B4	B3

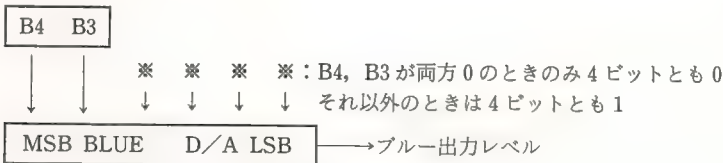
●グリーン：3ビット



●レッド：3ビット



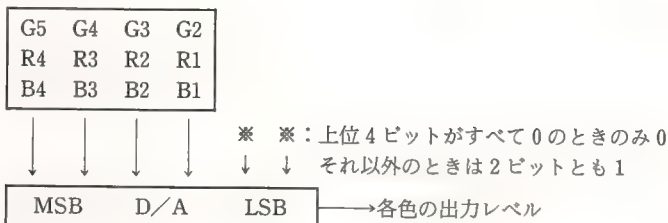
●ブルー：2ビット



(3) パレットによる4096色 (12ビット)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
G5	G4	G3	G2	—	—	R4	R3	R2	R1	—	B4	B3	B2	B1	—

●各色：4ビット



4.6.2 パレット

パレット指定画面の画像データは一旦パレット指定画面だけで合成されたあと、カラーパレットを参照することによってその実際の表示色が決定する。カラーパレットには各カラーコードに対応した実際の表示色が設定される。

カラーパレットは 16 個のパレットが 2 組あり、各パレットは 12 ビットで構成されている。このためカラーパレットを使用することにより基本モードでは 4096 色中 16 色を選択して表示を行うことができる。

各パレットは以下のように 12 ビットが 16 ビットポートに割り当てられている。

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
G5	G4	G3	G2	*	*	R4	R3	R2	R1	*	B4	B3	B2	B1	*

16 ビットが、

ビット 0～4 はブルー

ビット 5～9 はレッド

ビット 10～15 はグリーン

の各領域に分けられている。ただし、各領域の最下位 1 ビットもしくは 2 ビット（無効ビット、上記で*の場所）は現在意味を持たない。この無効ビットに相当する部分には前節で述べたのと同様に、RGB 各領域の有効ビットがすべて 0 のときは 0 に、それ以外の場合は 1 に設定された表示色が出力される。ソフトウェア的にもこれら無効ビットの設定は実際の出力と一致させた値にすることを推薦する。

カラーパレットは 16 本のパレットセットが 2 組あり、ポートアサインは次のようになっている。

パレットセット 0 ワードポート 0300H～031EH

パレットセット 1 ワードポート 0320H～033EH

(1) カラーパレットモード

カラーパレットの使用モードには以下の 4 種がある。

1) パレットセット 0 使用モード（カラーパレットモード 0）

合成画面はパレットセット 0 を使用して表示を行う。

2) パレットセット 1 使用モード（カラーパレットモード 1）

合成画面はパレットセット 1 を使用して表示を行う。

3) 混在モード（カラーパレットモード 2）

合成画面は基本的にパレットセット 0 を使用するが、指定された特定の 1 表示画面についてはパレットセット 1 を使用して表示を行う。

4) 同時 32 色モード（カラーパレットモード 3）

パレット指定画面（表示画面 0～3）にアサインされた 8 ビット／ピクセルのグラフィック画面に対して、パレットセット 0 及び 1 を同時に使用して 32 色の同時表示を行うことを指定する。その他の画面はパレットセット 0 を使用して表示される。パレット番号として使用しない上位 3 ビットは使われない。

カラーパレットモードの指定は V1/V2 モード対応の指定方法と V3 モード対応の指定方法がある。

① V1/V2 対応モード

PTLM2 (ワードポート 10CH ビット 8) = 0 のとき

ポート 32H ポート 31H

PLTM2	ビット 5	ビット 4	:	カラーパレットモード
0	0	0	:	カラーパレットモード 1
0	0	1	:	カラーパレットモード 2
0	1	0	:	カラーパレットモード 0
0	1	1	:	カラーパレットモード 2

② V3 対応モード

PTLM2 (ワードポート 10CH ビット 8) = 1 のとき

ポート 10H ポート 10H

PLTM2	ビット 7	ビット 6	:	カラーパレットモード
1	0	0	:	カラーパレットモード 0
1	0	1	:	カラーパレットモード 1
1	1	0	:	カラーパレットモード 2
1	1	1	:	カラーパレットモード 3

混在モード (カラーパレットモード 2) においてパレットセット 1 を参照する表示画面の指定は以下のポートによって指定される。

ワードポート 010CH

ビット 5	ビット 4	:	表示画面の指定
0	0	:	テキスト画面
0	1	:	スプライト画面
1	0	:	グラフィック画面 0
1	1	:	グラフィック画面 1

(2) カラーパレットのブリンク機能

2 組のパレットセットのうち片方だけを使用するモード (カラーパレットモード 0, 1) ではカラーパレットのブリンク機能が使用できる。これは 2 組のパレットセットを交互に切り換えて表示を行うことで実現している。

ブリンクの周期は、32 フレーム、64 フレーム、128 フレームから選択できる。またブリンクのデューティは、ON タイムを 12.5%、25%、50%、75% から指定できる。ON タイムにはカラーパレットモードで指定しておいたパレットセットが使用される。

カラーパレットのブリンク指定は以下のポートによって設定される。

ワードポート 010CH

ビット 3	ビット 2	:	ブリンクモード
0	0	:	ブリンク OFF
0	1	:	ブリンク周期 32 フレーム
1	0	:	ブリンク周期 64 フレーム
1	1	:	ブリンク周期 128 フレーム

ビット 1	ビット 0	:	ブリンクデューティ (ON タイム)
0	0	:	12.5%
0	1	:	25%
1	0	:	50%
1	1	:	75%

4.6.3 透明色

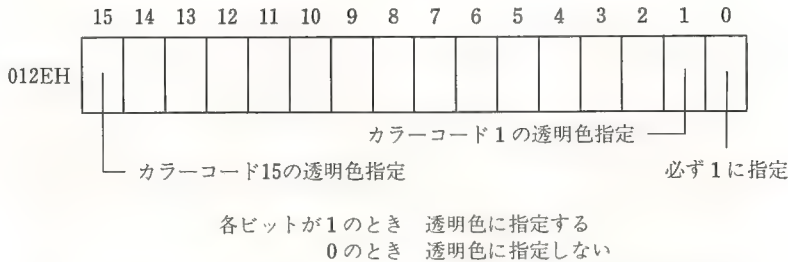
VA では画像ソースを合成するために、各表示画面に透明色が使用できる。通常低優先画面の画像はそれよりも高優先画面の画像に隠されて表示されないが、高優先画面の画像の透明色の部分には、低優先画面が表示される。透明色として扱う色には固定のものと指定できるものがある。

(1) パレット指定画面の透明色指定

パレット指定画面においては一部を除き任意のカラーコードを透明色として指定できる。透明色として指定されたカラーコードはパレットの値にかかわらず透明色として合成される。

(1.1) テキスト／スプライトの透明色指定

ワードポート 12EH で各カラーコードごとに透明色に設定するかどうかを指定する。ただしカラーコード 0 は設定にかかわらず必ず透明色として扱われる。

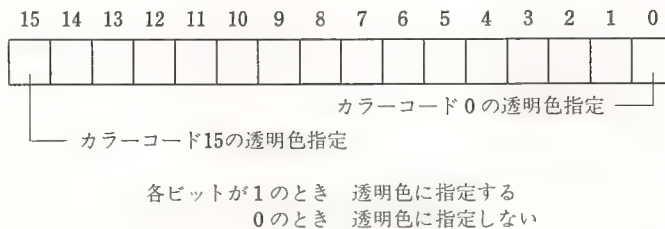


(1.2) グラフィック画面の透明色指定

グラフィック画面をカラーコード画面に割り当てたときの透明色の指定もテキスト／スプライトと同様である。グラフィック画面 0 はポート 124H で、グラフィック画面 1 はポート 126H で指定する。

ワードポート 124H (グラフィック画面 0 用)

ワードポート 126H (グラフィック画面 1 用)



ただし、カラーパレットモード 3 (32 色モード) でのカラーコード 16~31 は透明色に指定することはできない。

(2) 直接色指定画面の透明色指定

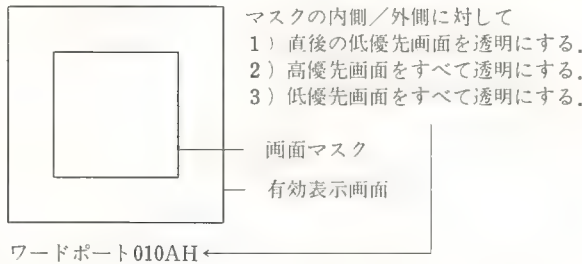
グラフィック画面を直接色指定画面に割り当てたときはカラー 0 (RGB とも 0) の画面は固定的に透明色として扱われる。その他の色を透明色に指定することはできない。

4.7 その他の機能

4.7.1 画面マスク機能

画面マスクは 1 枚の水平／垂直方向の長方形のマスクで水平／垂直方向ともにドット単位（垂直解像度が 400／408 ドットモードでは垂直方向は 2 ドット単位）に設定することができる。

この画面マスクを指定した優先表示画面の間に挿入することにより、挿入されたポジションよりも高優先または低優先の画面の表示をマスクする（透明色にする）ことができる。



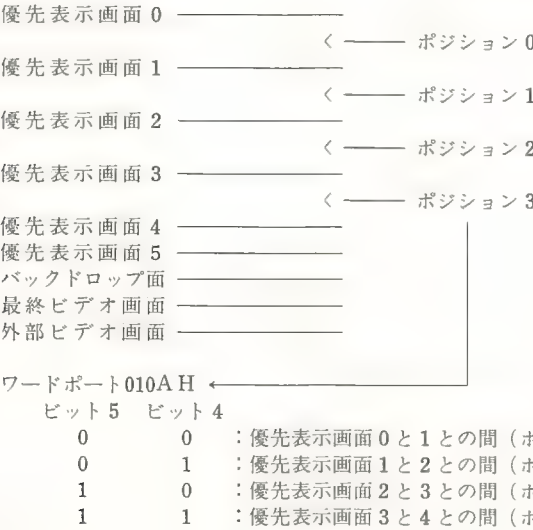
〈画面マスクの外側のオペレーション〉

ビット 3	ビット 2	
0	0	: ノーオペレーション
0	1	: 直後の低優先画面を透明にする
1	0	: 高優先画面をすべて透明にする
1	1	: 低優先画面をすべて透明にする

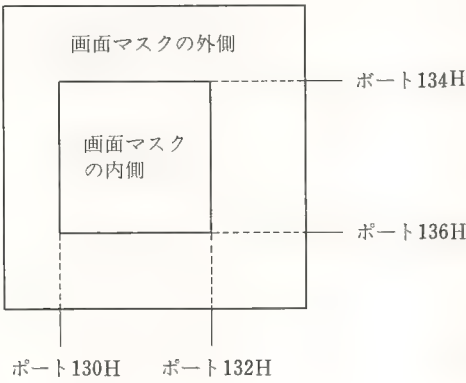
〈画面マスクの内側のオペレーション〉

ビット 1	ビット 0	
0	0	: ノーオペレーション
0	1	: 直後の低優先画面を透明にする
1	0	: 高優先画面をすべて透明にする
1	1	: 低優先画面をすべて透明にする

画面マスクの挿入位置は次の 4 カ所となっている。



画面マスクの座標設定はワードポート 130～136H で行う。



●画面マスク左右：640 ドット解像度の画面に対する値

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0130H	0	0	0	0	0	0	マスク 左端									
0132H	0	0	0	0	0	0	マスク 右端									

●画面マスク上下：200／204 ドット解像度の画面に対する値

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0134H	0	0	0	0	0	0	0	0	マスク 上端							
0136H	0	0	0	0	0	0	0	0	マスク 下端							

4.7.2 バックドロップ面

バックドロップ面は優先表示画面 5 の次に位置する表示画面であり、バックドロップカラーやボーダーカラーを指定する。

バックドロップ面には 1 色／4096 色が指定できる。色指定はパレットへの色設定と同様に RGB で行う。

●バックドロップ／ボーダーカラー指定

ワードポート 10EH

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
G3	G2	G1	G0	*	*	R3	R2	R1	R0	*	B3	B2	B1	B0	*

[*]はパレットと同様に RGB 各領域の上位ビットがすべて 0 のときに、それ以外のときは 1 に設定する。

バックドロップ面の表示モードには 4 種のモードがある。ただし 24.8KHz の CRT 使用時ではバックドロップモード 0 のみ指定可能である。

- 1) バックドロップモード 0
有効表示領域：バックドロップカラー
ボーダー領域：透明
- 2) バックドロップモード 1
有効表示領域：透明
ボーダー領域：バックドロップカラー
- 3) バックドロップモード 2
有効表示領域：バックドロップカラー
ボーダー領域：バックドロップカラー
- 4) バックドロップモード 3
有効表示領域：透明
ボーダー領域：透明

●バックドロップモードの指定

ワードポート 10CH

ビット 13 ビット 12

0	0	:	バックドロップモード 0
0	1	:	バックドロップモード 1
1	0	:	バックドロップモード 2
1	1	:	バックドロップモード 3

24.8KHz 使用時ではモード 0 のみ指定可能。

4.7.3 表示禁止機能

(1) ビデオサプレス機能

この機能を用いるといかなる表示モードにおいても表示を禁止することができる。

ワードポート 100H

ビット 13 (XVSP)

0	: 表示禁止モード
1	: 表示モード

(2) グラフィック表示禁止機能

この機能ではすべてのグラフィック画面表示が禁止される。また表示のための GVRAM へのアクセスもすべて行われなくなる。GVRAM への描画機能とは無関係である。

ワードポート 100H

ビット 15 (GDEN0)

0	: グラフィック表示機能禁止
1	: グラフィック表示機能許可

ワードポート 100H

ビット 5 (GDEN1)

0	: グラフィック表示回路リセット
1	: グラフィック表示回路動作

(注)GDEN1によりグラフィック表示回路にリセットをかけると、ポートの一部が初期化される。また、このビットの操作シーケンスは4.9.5を参照すること。

4.7.4 同期信号マスク

この機能により水平同期信号の出力を禁止することができる。これは電源投入時に TSP (IDP) の初期設定が行われるまでの間、水平同期信号の出力を禁止して CRT に不確定な水平同期信号が入力されることを防ぐために使用される。

ワードポート 100H
 ビット 12 (SYNCEN)
 0 : 水平同期信号出力禁止
 1 : 通常表示状態

システム起動時の CRT 同期の取り方。

- ①リセットにより表示禁止／同期出力禁止となる
- ② TSP (テキスト／スプライトプロセッサ) を初期設定する
- ③水平同期出力 ON
- ④ 1 秒以上ウエイト
- ⑤表示開始

4.8 文字セット／文字コード／文字フォント

VA には ANK・JIS 第一／第二水準の漢字・拡張漢字・ユーザ定義文字のフォントを格納したキャラクタジェネレータ ROM／RAM(一括して CGROM と呼ぶ)が搭載されている。テキスト表示時にコントローラは、TVRAM に格納されている文字コードにより CGROM から文字フォントを読み出して表示する。また CGROM は CPU から読み出すこともでき、グラフィック画面にビットマップとして文字を表示することができる。

4.8.1 文字セット

(1) 1 バイト文字

PC シリーズで使用している ANK 文字セット 256 文字

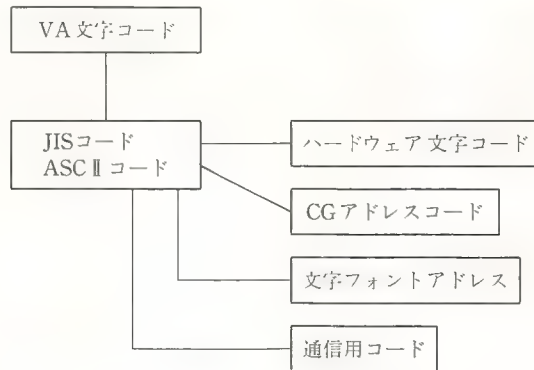
(2) 2 バイト文字

PC-9800 シリーズと同様の下記の文字セット

JIS 非漢字(全角)	453 文字
NEC 非漢字(全角／半角)	443 文字
JIS 第 1 水準漢字	2965 文字
JIS 第 2 水準漢字	3384 文字
ユーザ定義文字	188 文字

4.8.2 文字コード体系

VA で文字を扱う場合のコード体系は次のようになっている。

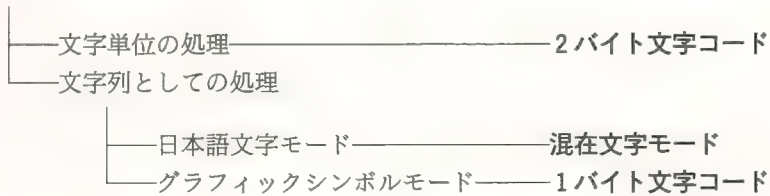


VA システム内部の文字(列)はソフトウェア的には VA 文字コードで表されている。この VA 文字コードから変換された JIS コードをピボットとしてハードウェア文字コード／CG アドレスコード／文字フォントアドレス／通信用コードに相互変換される。

(1) VA 文字コード

VA 文字コードとして 2 バイト文字コード／混在文字コード／1 バイト文字コードの 3 種のコードがあり、場合によって使いわける。

VA 文字コード



(1.1) 2 バイト文字コード

文字を文字単位に処理する場合に使用するコードであり、すべての文字を 2 バイトで表す。

2 バイト文字——シフト JIS コード

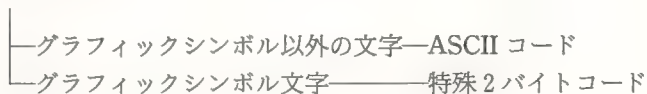
1 バイト文字——上位バイト = 0 下位バイト = ASCII コード

(1.2) 混在文字コード

日本語文字列を処理する場合に使用するコードであり、各文字は 1 バイトまたは 2 バイトで表す。

2 バイト文字——シフト JIS コード

1 バイト文字



グラフィックシンボル文字とは1バイトコード(ASCIIコード)で、

81～9FH E0～FCH

を指す。これらの文字はシフト JIS コードの第1バイトと同じコードであるため1バイトでは表せない。したがって次のような2バイトコードで表す(数字は16進)。

1バイト文字コード ↔ 混在文字コード

81	84C1
82	84C2
⋮	⋮
9F	84DF
E0	84E0
⋮	⋮
FC	84FC

(1.3) 1バイト文字コード

1バイト文字のみを使用するコードであり、日本語文字は扱えない。

1バイト文字——ASCIIコード

(1.4) シフト JIS コード

シフト JIS コードは PC-9800 シリーズ MS-DOS で使用されているシフト JIS コードと同様であり、2バイト文字コード／混在文字コードで2バイト文字を表すのに使用される。JIS コードとの対応は次のようになっている。

JIS コード		シフト JIS コード
2121～5E7EH	↔	8140～9FFCH
5F21～7E7EH	↔	E040～EFFCH
〈拡張文字コード〉	↔	F040～FCFCH

●シフト JIS コードと JIS コードの対応

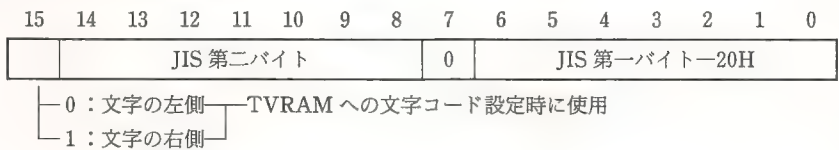
		————→第2バイト													
		40	41	⋮	7E	7F	80	⋮	9D	9E	9F	A0	⋮	FB	FC
↓ 第1 バイト	81	2121	2122	⋮	215F	--	2160	⋮	217D	217E	2221	2222	⋮	227D	227E
	82	2321	2322	⋮	235F	--	2360	⋮	237D	237E	2421	2422	⋮	247D	247E
	⋮			⋮				⋮					⋮		
	9F	5D21	5D22	⋮	5D5F	--	5D60	⋮	5D7D	5D7E	5E21	5E22	⋮	5E7D	5E7E
	E0	5F21	5F22	⋮	5F5F	--	5F60	⋮	5F7D	5F7E	6021	6022	⋮	607D	607E
	⋮			⋮				⋮					⋮		
	EF	7D21	7D22	⋮	7D5F	--	7D60	⋮	7D7D	7D7E	7E21	7E22	⋮	7E7D	7E7E
	F0	〈User Defined Extension〉													
	⋮	⋮													
	FC	〈User Defined Extension〉													

(2) ハードウェア文字コード

ハードウェア文字コードとは VA システム内のハードウェアに対して文字を指定するために使用されるコードであり、TVRAM に格納して文字表示を行う場合と、CGROM アクセスポート（ワードポート 14CH）より文字フォントをアクセスする場合に使用される。

ハードウェア文字コードは 16 ビットで構成される。

● 2 バイト文字（日本語文字）



● 1 バイト文字（ANK 文字）



(3) CG アドレスコード

CG アドレスコードとは V1/V2 モード専用漢字アクセスポートより CGROM をアクセスする場合に使用される。通常 V1/V2 モード時のみ使用される。

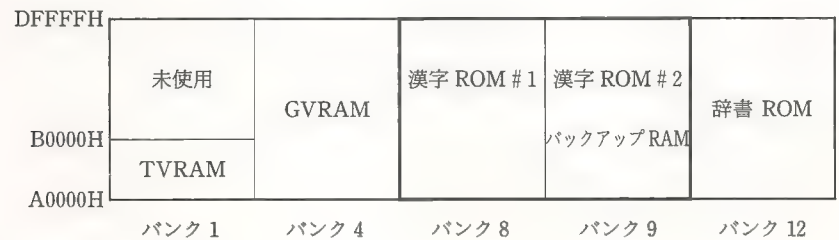
(4) 文字フォントアドレス

CGROM を CPU のメモリ空間にマッピングしてアクセスする場合の文字フォントが存在するアドレスである。文字フォントアドレスの求め方は次節を参照。

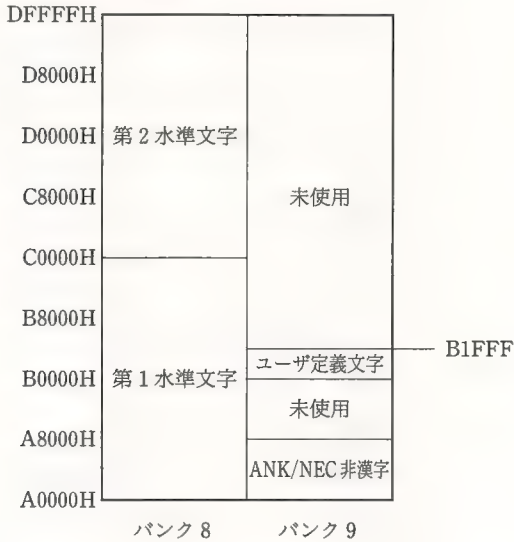
4.8.3 文字フォント

(1) メモリ空間からのアクセス

VA の文字フォントは CPU から見た場合、システムメモリエリア (A0000~DFFFFH) の 2 つのバンク（バンク #8/#9）に格納されている。バンク 8 には JIS 漢字／非漢字が、バンク 9 には NEC 非漢字／ANK 文字／ユーザー定義文字が割り当てられている。

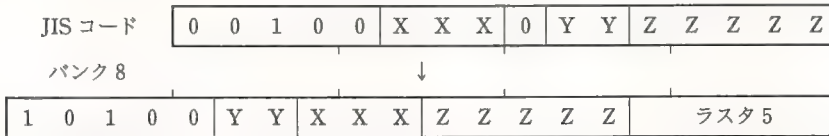


●バンク 8／9 の詳細

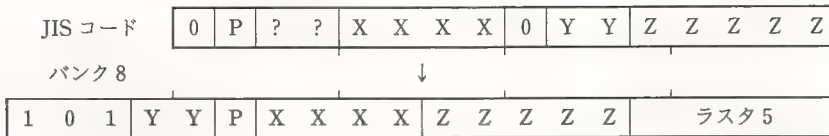


JIS コード／ASCII コードから文字フォントアドレスへの変換は文字種によって 8 種類に分けられる。

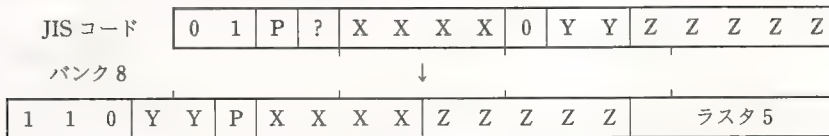
① JIS 非漢字 2021～2771H



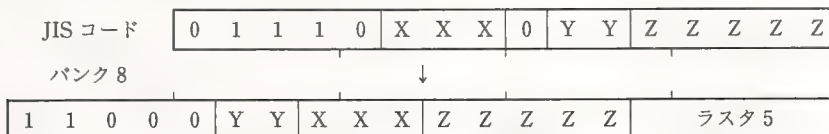
② JIS 第 1 水準漢字 3021～4F53H



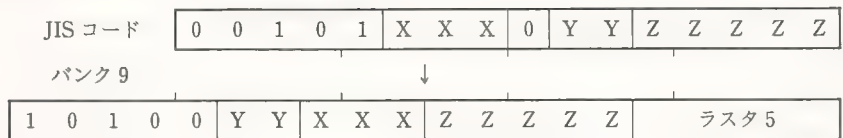
③ JIS 第 2 水準漢字 (I) 5021～6F7EH



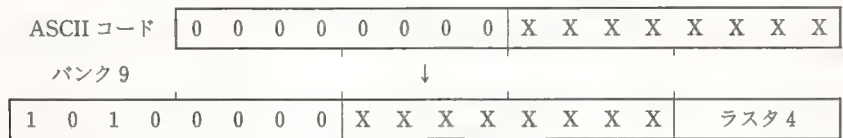
④ JIS 第 2 水準漢字 (II) 7021～733FH



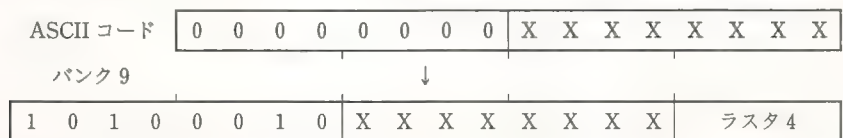
⑤ NEC 非漢字 2820～2F7FH



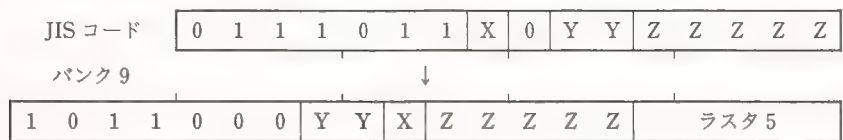
⑥ ANK 文字(8×16 フォント) 0000～00FFH



⑦ ANK 文字(8×8 フォント) 0000～00FFH



⑧ ユーザ定義文字 7620～773DH



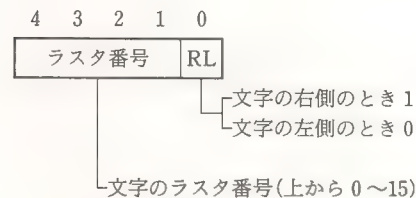
●ラスト

ラスト 5 / ラスト 4 は文字フォントの各ラストを指定するためのフィールドであり、5 ビットまたは 4 ビットからなる。

〈例：「漢」〉

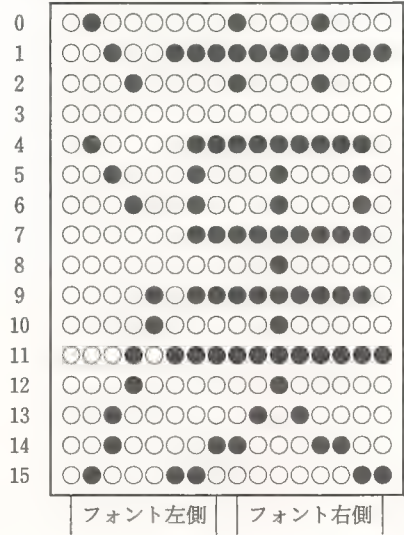
①ラスト 5

全角文字(16×16 ドット)のラスト番号とフォントの左右を指定するもの。ただし、NEC 非漢字半角(JIS コード 2921H～2B7EH)は文字の左側のみを使用して 8×16 ドットフォントを取り出す。



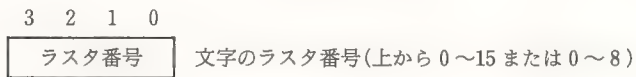
※必ず左右別々にバイトアクセスすること。
ワードアクセスはできない。

ラスト番号



②ラスト 4

ANK 文字 (8×16 または 8×8 ドット) のラスト番号を指定するもの。



ラスト 5 のときと同じくバイトアクセスしかできない。

(2) I/O 空間からのアクセス

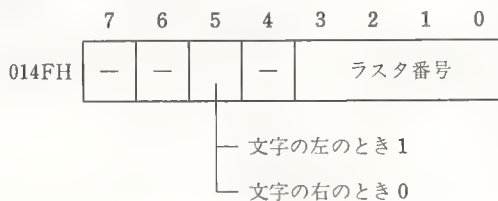
V3 モードではポート 14C～14FH を用いて I/O 空間から文字フォントを読み出すことができる。この場合文字の指定にはハードウェア文字コードを用いる。

●文字とラストアドレスの指定

OUT 14CH: ハードウェア文字コード下位バイト

OUT 14DH: ハードウェア文字コード上位バイト
(ワードアクセス可能)

OUT 14FH: ラスト番号/フォント左右



ANK 文字/半角文字は左側だけ使用する。

●フォントデータの読出し

IN 14EH (バイトアクセスのみ): フォントデータ

外字キャラクタージェネレータには書込みも可能

4.9 CRT 走査モードと表示解像度

VA においては各種の CRT ディスプレイをサポートするために以下に示す走査モードがある。

4.9.1 使用できる CRT ディスプレイ

VA がサポートする CRT ディスプレイを以下に示す。

- | | |
|-------------------|-------------|
| 1) 水平周波数 15.98KHz | ノンインターレース表示 |
| 2) 水平周波数 15.73KHz | インターレース表示 |
| 3) 水平周波数 24.8KHz | ノンインターレース表示 |

4.9.2 走査モードの設定

使用する CRT ディスプレイとその走査モードの設定方法を以下に示す。

(1) ディップスイッチによる設定

使用する CRT ディスプレイの種類をディップスイッチにより設定する。

SW1	: CRT モード
ON	: 水平周波数 24.8KHz CRT
OFF	: 水平周波数 15KHz CRT

(2) 表示制御ポートによる設定

インターフェイスの走査モードを表示制御ポートにより設定する。

ポート 100H の RSM1/RSM0

ビット 7 ビット 6

0	0	: ノンインターレースモード 0
0	1	: ノンインターレースモード 1
1	0	: インターレースモード 0
1	1	: インターレースモード 1

ビット 7 (RSM1) = 1 が設定された場合には、ディップスイッチ SW1 が ON に設定されている場合でも、水平周波数 15KHz のタイミングが選択される。現在の水平周波数の設定は次のポートで確認できる。

ポート 1CBH	ビット 3
0	15KHz モード
1	24.8KHz モード

4.9.3 水平解像度の設定

●テキスト

ポート 30H ビット 0

0	: 40 文字/行
1	: 80 文字/行

●スプライト

常に 640 ドット

●グラフィックス

ポート 102H (ワード)

ビット 4 グラフィック画面 0 水平解像度

0	: 640 ドット
1	: 320 ドット

ビット 12 グラフィック画面 1 水平解像度

0	: 640 ドット
1	: 320 ドット

4.9.4 垂直解像度の設定

(1) ノンインターレースモード

ノンインターレースモードにおいては、インターレース(飛越し走査)を行わずに表示される。

15KHz CRT 使用時に 400/408 ライン表示はできない。

①ノンインターレースモード 0

24.8KHz CRT を使用し 200/204 ライン表示を行う場合には、奇数ラスタはブランキングとなる。

②ノンインターレースモード 1

24.8KHz CRT を使用し 200/204 ライン表示を行う場合には、ノンインターレースモード 0 と異なりペアの偶数奇数ラスタに同一データが表示される。

このモードでは垂直解像度の設定は次のようになる。

●テキスト

TSP の SYNC コマンドのパラメータで設定 (4.4.2 参照)

●スプライト

TSP の SPRON コマンドでの縦ズームパラメータで設定

200/204 ライン：

15KHz でテキストが 200/204 ラインのとき：ノーマルモード

それ以外：垂直拡大モード

400/408 ライン：

テキストが 200/204 ラインのとき：設定不可

テキストが 400/408 ラインのとき：ノーマルモード

●グラフィックス

ポート 100H ビット 1,0 で設定

ビット 1 ビット 0

0 0 : 400 ライン

0 1 : 408 ライン

1 0 : 200 ライン

1 1 : 204 ライン

(2) インターレースモード

インターレースモード 0 においては、NTSC 同期信号を用いてインターレース表示を行う。15KHz CRT のみ使用可能である。

①インターレースモード 0 (200/204 ライン用)

このモードでは 200/204 ラインの画面を偶数フィールド/奇数フィールドとも同一データとして表示する。400/408 ライン表示はできない。

②インターレースモード 1 (400/408 ライン用)

このモードでは 400/408 ラインの画面の偶数行を偶数フィールドに、奇数行を奇数フィールドに分けて交互に表示する。

このモードにおける垂直解像度の設定は以下のようになる。

●テキスト

TSP の SYNC コマンドのパラメータで設定する (4.4.2 参照)。

●スプライト

TSP の SPRON コマンドでの縦ズームパラメータで設定

200/204 ライン：ノーマルモード

400/408 ライン：設定不可

●グラフィックス

ポート 100H ビット 1,0 で設定

ビット 1 ビット 0

0 0 : 400 ライン

0 1 : 408 ライン

1 0 : 200 ライン

1 1 : 204 ライン

4.9.5 同期モードの切換え手順

VA では 24.8KHz と 15KHz をソフトウェアにより切り換えることができる。ただしその際には以下に示す手順でモードの遷移を行う必要がある。そうでない場合の GVRAM の内容は保証されない。また、この手順はノンインターレースとインターレースの切換え、インターレース時の内部同期／外部同期の切換え(4.10 ビデオディジタル参照)時にも使用される。

- ① ワードポート 100H のビット 15(GDEN0) = 0, ビット 14(XVSP) = 1, ビット 12(SYNCEN) = 0 に設定してグラフィック表示機能を禁止する。
- ② ポート 40H のビット 5(VRTC)が 0 → 1 → 0 → 1 と変化するタイミングまで待つ。
または VRTC 割込みを少なくとも 2 回待ち合わせる。
- ③ ワードポート 100H のビット 5(GDEN1) = 0 に設定してグラフィック表示回路をリセットする。これにより設定されている各パラメータもリセットされる。
- ④ ワードポート 100H のビット 7/6(RSM1/0), ビット 4(SYNCM)を設定する。
- ⑤ ワードポート 100H のビット 5 (GDEN1) = 1 に設定してグラフィック表示回路のリセットを解除する。
- ⑥ 新表示モードに従って、すべての表示パラメータを再設定する(注 2)。
- ⑦ ワードポート 100H のビット 15(GDEN0) = 1 に設定してグラフィック表示機能を再開する。
- ⑧ ワードポート 100H のビット 12(SYNCEN) = 1 に設定して水平同期信号を出す。
- ⑨ 少なくとも 1 秒待ち、ワードポート 100H のビット 14(XVSP) = 0 に設定して表示を再開する。

(注 1) 上記各項目は独立した I/O 命令として発行しなければならない。

例えば、④と⑤とは同一ポート内にあるため、1 回の OUT 命令で処理することができるが、そのようにせずに、④項で 1 命令、⑤項で 1 命令というように独立に設定する必要がある。

(注 2) リセットされるため再設定が必要なポートを以下に示す。

ワードポート 100H

ワードポート 102H

ワードポート 106H

ワードポート 110H

4.10 ビデオディジタイズ

外部ビデオ信号を入力し、GVRAM にリアルタイムで取り込むビデオディジタイズ機能がサポートされている。このとき画像出力には入力された画像が表示される。この機能を使用するためにはオプションのビデオボードの接続が必要になる。

4.10.1 ビデオディジタイズモードへの設定

ビデオディジタイズ機能は以下の動作モード時にのみ使用可能となる。

ワードポート 100H

GDEN0=1	:	グラフィック機能イネーブル
GDEN1=1	:	グラフィック機能イネーブル
DM=1	:	シングルプレーンモード
RSM1=1	:	インターレースモード
SYNCM=1	:	外部同期モード
GVM=1	:	ビデオディジタイズモード

(注1) ビデオディジタイズモードで使用できるグラフィック画面はグラフィック画面 0 のみである。

(注2) RSM1=1 を設定する必要があるため、15KHz CRT でないと表示されない。

● NTSC 同期信号の切換え

VA は NTSC 同期信号を発生する回路を内蔵しており、インターレース動作はこの内部同期信号を用いて行う場合と、外部から入力される同期信号を用いて行う場合とがある。外部ビデオ信号をフレームディジタイズする場合や、外部ビデオ信号に、VA の画面をスーパーインポーズする場合には、外部からの同期信号を用いる。

ポート 100H

ビット 4(SYNCM)

0	:	内部同期モード
1	:	外部同期モード

4.10.2 ビデオディジタイズ時の取込みフォーマット制御

ビデオディジタイズ時の取込みフォーマット制御として下記の機能がある。

(1) インターレースモードと垂直解像度

・インターレースモード 0

200/204 ラインのみが設定可能であり、入力信号の偶数フィールド/奇数フィールドともに同一アドレスの GVRAM に取り込まれる。このため GVRAM に格納されるビデオデータはフリーズされたタイミングによって偶数フィールドまたは奇数フィールドのいずれか一方のフィールドのデータとなる。

・インターレースモード 1

400/408 ラインのみが設定可能であり、入力信号の偶数フィールド/奇数フィールド両方が GVRAM に取り込まれる。

(2) 水平解像度

ビデオディジタイズモードにおいても 640/320 ドットが選択できる。

(3) ピクセルサイズ

ビデオディジタイズモードにおけるピクセルサイズは、1 ビット/ピクセルを除いたピクセルサイズが設定可能である。この場合各サイズにおいて外部ユニットによって作成されたデジタルビデオデータのうち次のビットが有効となる。

4 ビット/ピクセル——下位 4 ビット

8 ビット/ピクセル——下位 8 ビット

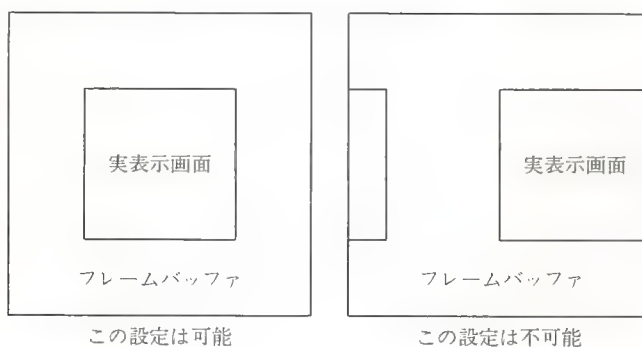
16 ビット/ピクセル——16 ビット

(4) フレームバッファ

ビデオディジタイズモードにおいても GVRAM のアドレスはフレームバッファを用いて決定されるが、その設定には以下の制限がある。

(4.1) 実表示画面とフレームバッファとの関係

実表示画面が水平方向のラップアラウンドをしないように設定する必要がある。



(4.2) ドットアドレス

ドットアドレスは常に 0 でなければならない。

4.10.3 ビデオディジタイズ動作

ビデオディジタイズ動作は上記各モードの設定後、ワードポート 100H ビット 14(GVM)を 1 にすることにより開始される。GVM ビットは垂直同期信号によりサンプリングされている。

GVM=1 が設定されると、GVRAM へのビデオデータの取込みが開始されると同時に、ディジタイズされている画面が表示される。

この後 GVM=0 にすることによって、そのときディジタイズ中のフィールドが終了した時点で GVRAM のデータをフリーズして表示モードに戻る。

4.10.4 ビデオディジタイズモードにおける表示

ビデオディジタイズモードにおいても表示機能はアクティブになっているが、グラフィック表示機能については GVRAM の表示のための読出しが行われずディジタイズ中のグラフィック画面が表示される。

このモードにおいてはグラフィック画面0のみが有効な画面となっており、グラフィック画面1は表示されないように制御ポートを設定しておく必要がある。

なお、その他の表示画面(テキスト画面、スプライト画面)は通常の表示モードと同様に表示可能であり、デジタルサイズ中であるグラフィック画面0もそれらと正常に合成されて表示される。

また、グラフィック画面0用のフレームバッファ(#0, 2, 3)はいずれもデジタルサイズ動作が可能なため、たとえばフレームバッファ0を使用してビデオデジタルサイズを行う場合には、その他のフレームバッファ(#2, 3)が動作しないように分割画面の高さ(ポート 252H, 272H)を0にしておく必要がある。

第5章 その他のハードウェアコントロール

この章では表示回路以外の各種ハードウェアの制御方法について記載しているが、制御ポートのビットの意味を詳細には解説していない。したがって、この章は第2章のI/Oポートマップを常に参照しながら読む必要がある。

5.1 スイッチとシステム動作モード

5.1.1 モードスイッチ

(1) モードスイッチ概要

VAにはモードスイッチが3種あり、これらのモードスイッチの設定により、システムの起動時の動作モードを指定することができる。

- ①システムモードスイッチ
- ②ディップスイッチ
- ③スピードスイッチ(このスイッチはソフトウェアでテストできない)

(2) システムモードスイッチ

システムモードスイッチは2ポジションのスライドスイッチであり、BASICのモードなど、VAシステム全体の動作モードを設定する。



このスイッチで設定するのはV1モードとV2モードだけであり、V3モードは一旦V1/V2モードで起動した後、ソフトウェア的に移行する。したがってV3モード用モード設定スイッチはない。V3モードへの移行はポート153Hで行う。

システムモードスイッチの設定はリセット時にシステム内に取り込まれ、その後は以下のポートで読み出せる。

IN 150H

ビット1	ビット0	: システムモード
0	0	: 予約(未使用)
1	0	: V1モード
0	1	: V2モード
1	1	: 予約(未使用)

システムが起動した後、3種類の動作モードをポート1CDHによってLEDに表示する。

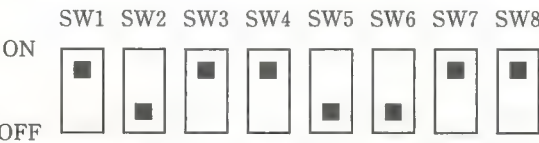
OUT 1CDH(システムポート 8)

ビット 6	ビット 5	ビット 4	
1	1	0	V1 モード
1	0	1	V2 モード
0	1	1	V3 モード

(注) ポート 1CFH により各ビット単位にセット／リセットを行うことができる。

(3) ディップスイッチ

ディップスイッチは 8 種類の各種モード設定スイッチにより構成されている。これらのモードスイッチの一部は直接ハードウェアを制御し、他はソフトウェアにより読み出されてシステム起動時の動作モードを決定する。



スイッチ	機能	設定	指定モード
SW 1	CRT モード (ディスプレイの種類)	ON	24 KHz タイプ
		OFF	15 KHz タイプ
SW 2 SW 5	予 約		
SW 6	RS-232C 外部同期／内部同期	ON	外部同期 モード
		OFF	内部同期 モード
SW 7	内蔵 GDD ポートモード	ON	ブートする
		OFF	ブートしない
SW 8	CRT クロック出力	ON	外部に出力しない
		OFF	外部に出力する

* SW 6.8 を除くスイッチの値は次のポートで読み込める

SW 1	1CBH, 40H
SW 2~5	1C9H
SW 7	40H

(4) スピードスイッチ

●概要

本機 CPU で μ PD780 の命令を実行する場合の動作スピードは、88M/F シリーズで用いられている μ PD780 の動作スピードと全く同一なわけではない。したがって、本機が V1/V2 モードで動作している場合に 88M/F シリーズとの動作速度の差を修正して、アプリケーションソフトウェアを正常に動作させるためのスピード調整回路が内蔵されている。この回路を使用するかどうかの設定を行うのがスピードスイッチである。

スピード調整の方法としては、一定の期間に対してある一定の数のウェイトサイクルを挿入することにより平均的な速度を制御する方式を採用する。これは次のような理由による。

- ①各命令の処理クロック数は一律に μ PD780 との差があるわけではない。
- ②一部の I/O ポートは、CPU の I/O トラップ処理によりアクセスされるためその場合 88M/F シリーズでは生じないオーバーヘッドが発生する。
- ③ VRAM をアクセスする場合に挿入されるウェイトの数も 88M/F シリーズの場合とは異なる。

●スピードモード

・モードの設定

スピードスイッチは 2 ポジションのスライドスイッチである。

- | | | | | |
|-------|---|--------------|---|-----------|
| 2 モード | : | μ PD9002 | 8MHz の動作速度 (ハイスピードモード) | } の等価動作速度 |
| 1 モード | : | μ PD780 | 88R シリーズの H モード
88H シリーズ 4MHz 時の H モード | |

このスイッチの状態は、リセット信号がアクティブとなっている時にサンプリングされるため、動作中にスピードを切り換えることはできない。

また、V3 モードでは常にハイスピードモードで動作し、スイッチに影響されない。

・スピードモードの読出し

現在のスピードモードはシステムポート 6 から読み出すことができる。

IN 1C9H	ビット 5
0	: 1 モード (標準スピードモード)
1	: 2 モード (ハイスピードモード)

5.1.2 メモリスイッチ

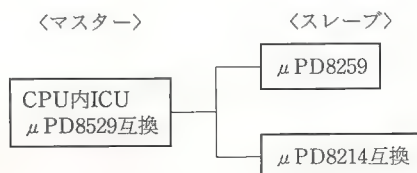
外字 RAM キャラクタジェネレータとして使用されているバッテリーバックアップメモリのうち、B1FC2H/B1FC6H はメモリスイッチとして IN 命令で読み出すことができる。このメモリスイッチは V1/V2 モードにおいて 88M/F シリーズのディップスイッチの代わりとなる。またメモリとしてアクセスする場合と異なり、IN 命令ではメモリモードにかかわらずにリードのみが可能である。

IN 30H : B1FC2H 番地のメモリスイッチ

IN 31H : B1FC6H 番地のメモリスイッチ

5.2 割込み

VA の割込みを処理するハードウェアの構成は次のようになっている。



CPU 内 ICU をマスタとして、2 種類のスレーブがマスタのレベル 7 割込みに接続されている。この内どちらのスレーブを使用するかによって 2 つのモードがあり、V3 モードと V1/V2 モードに対応している。

① 8259 モード V3 モード対応

スレーブとして μ PD8259 を使用する。CPU 内 ICU と合わせて 15 レベルの割込みを処理することができるが実際にはそのうちの 13 レベルが使用できる。

② 8214 モード V1/V2 モード対応

スレーブとして μ PD8214 互換ハードウェアを使用する。

CPU 内 ICU は実際には働かず、すべてをスレーブにまかせるため、88M/F シリーズと互換性のある割込み処理が行われる。

割込みレベルは 8 レベル。

リセット時の割込みモードは 8214 モードとなっている。8259 モードにするためにはポート 158H を使用する。

OUT 158H (出力データはダミー)

ポート 158H に出力することによって、割込みモードは 8259 モードになる。ただし、一度 8259 モードに変更すると、リセット以外に再び 8214 モードにすることはできない。

●モード変更時の注意点

- ①メイン CPU を割込み禁止状態にすること
- ②各割込み要求を止めてから行うこと
- ③モード変更を行った後は、すみやかにスレーブ μ PD8259, CPU 内 ICU をこの順序で初期設定すること
- ④初期設定が終ると割込み要求を再開できる

● μ PD8259 概説

8259 には 2 つのポートがあり、それぞれポート 0、ポート 1 と呼ぶことにする。各々のポートにコマンドを送って動作を制御する。コマンドには次の 5 種類がある。

① μ PD8259 の初期設定(ICW)

OUT ポート 0, ICW1

OUT ポート 1, ICW2

(OUT ポート 1, ICW3 ICW1 で必要かどうか指定する)

(OUT ポート 1, ICW4 ICW1 で必要かどうか指定する)

②インターラプトマスク制御(OCW1)

OUT ポート 1, OCW1 インターラプトマスク設定

IN ポート 1 インターラプトマスク読出し

③割込み終了／優先回転制御(OCW2)

OUT ポート 0, OCW2

④割込みステータス制御／読出しレジスタ指定(OCW3)

OUT ポート 0, OCW3

⑤IRR/ISR/ポーリングモード読出し

IN ポート 0 どのレジスタを読み出すかは OCW3 で指定する

ポートアドレスは以下のようになっている。

	マスタ	スレーブ
ポート 0	188H	184H
ポート 1	18AH	186H

5.2.1 8259 モード (V3 モード)

このモードでは発生した割込みはすべて V30 用割込みとして処理される。

(1) 割込みレベルと用途

割込みレベル	機 能	割込みベクタ番号
IR0	タイマ 1 割込み	08H
IR1	キーボード割込み	09H
IR2	VRTC 割込み	0AH
IR3	UINT0 (Bus # 24)	0BH
IR4	RS-232C 割込み	0CH
IR5	UINT1 (Bus # 25)	0DH
IR6	UINT2 (Bus # 26)	0EH
IR7	スレーブ	——
IR7	SGP 割込み	10H
IR9	ハードディスク (Bus # 27)	11H
IR10	UINT4 (Bus # 28)	12H
IR11	FDC 割込み	13H
IR12	サウンドコントローラ	14H
IR13	汎用タイマ 3 (マウスタイマ)	15H
IR14	リザーブ	16H
IR15	リザーブ	17H

(2) I/O ポートアドレス

	ポート 0	ポート 1
ICU (マスタ)	188H	18AH
μPD8259 (スレーブ)	184H	186H

(3) 割込みマスクポート

●コントローラによるマスク

μPD8259 に対して各割込み入力を独立に禁止する機能がある (OCW1)。

・マスタ (ICU)

18AH	IR7M	IR6M	IR5M	IR4M	IR3M	IR2M	IR1M	IR0M
------	------	------	------	------	------	------	------	------

・スレーブ(μPD8259)

186H	IR15M	IR14M	IR13M	IR12M	IR11M	IR10M	IR9M	IR8M
------	-------	-------	-------	-------	-------	-------	------	------

機能

IRnM	:	割込みマスク
0	:	割込み許可
1	:	割込み禁止

● RS-232C マスク

8259 モードでは RS-232C インターフェイスのコントローラ(USART)用の割込み入力が1レベルであるが、USART からの割込み要因には以下の3種類のものがある。

- ①受信バッファ レディ
- ②送信バッファ レディ
- ③送信バッファ エンプティ

このため、各割込み要因に対して独立にマスクを施すポートがある。

OUT 1CDH (システムポート8)

ビット2: RS-232C TXRDY 割込みマスク

0: 割込み禁止

1: 割込み許可

ビット1: RS-232C TXEMPTY 割込みマスク

0: 割込み禁止

1: 割込み許可

ビット0: RS-232C RXRDY 割込みマスク

0: 割込み禁止

1: 割込み許可

(注) ポート 1CFH により各ビット単位にセット/リセットを行うことができる。

●サウンドコントローラ割込みマスク

ポート 32H のビット7によってサウンドコントローラからの割込みを独立してマスクできる
[5.10 サウンドインターフェイス]参照

5.2.2 8214 モード(V1/V2 モード)

本モードにおいて割込みコントローラは以下の2種類の割込みベクタを発生させることができる。

- ①μPD780 用の割込みベクタ
- ②V30 用の割込みベクタ

つまり8214 モードにおいては、メイン CPU が μPD780 の命令コードを実行中に発生した割込みに対しては、μPD780 の割込みベクタを発生し μPD780 と等価の割込み処理を行い、メイン CPU が V30 の命令コードを実行中に発生した割込みに対しては、V50 の割込みベクタを発生し V30 としての割込み処理をおこなう。このため割込みコントローラとしては、両タイプの割込みベクタを発生する必要がある。

この両ベクタは割り込みコントローラが割り込み発生時のメイン CPU 動作モードを判別して自動的にその選択を行う。

(1) 割り込みテーブル

割り込みレベル (名称)	機能	割り込みベクタ番号	
		V30	μPD780
INT0	RS-232C 受信割り込み	40H	00H
INT1	VRTC 割り込み	41H	02H
INT2	汎用タイマ 2 割り込み	42H	04H
INT3	UINT0 (Bus # 24)	43H	06H
INT4	サウンドコントローラ	44H	08H
INT5	UINT1 (Bus # 25)	45H	0AH
INT6	汎用タイマ 3 割り込み	46H	0CH
INT7	SGP 割り込み	47H	0EH

(2) 8214 モードにおける ICU／8214 互換ハードウェアの初期設定

● ICU

ICW1	0	0	0	1	0	0	0	1
ICW2	0	0	0	0	0	0	0	0
ICW3	1	0	0	0	0	0	0	0
ICW4	0	0	0	0	0	0	1	1
OCW1	0	1	1	1	1	1	1	1 (スレーブ以外はマスク)

● 8214 互換ハードウェア

OUT E4H, 7

(3) 割り込みコントロール回路制御ポート

OUT／IN E4H

- ビット 3 : カレントステータスレジスタ制御
 - 0 : 有効(参照されて発生)
 - 1 : 無効(入力優先順位のみで発生)
 - ビット 2～0 : カレントステータスレジスタ
- 現在の割り込みレベルを OUT する。

(4) 8214 モード割り込み処理方法

①イニシャライズ

OUT 0E4H, 7

②割込み手続き

- 1) レジスタをセーブする。
- 2) コマンドポートに現在設定されているデータをシステム共通領域のポート値保存領域から読出しスタックにセーブする。
- 3) DI 状態で行うべき割込みサービスを行う。
- 4) 発生した割込みレベルのカレントステータスレジスタの制御モードをコマンドポートとシステム共通領域にライトする。
- 5) EI 命令を実行。
- 6) EI 状態で行うべき割込みサービスを行う。
- 7) 2)の処理でスタックにセーブしたこの割込み発生以前にコマンドポートに設定されていたコマンドをリストアする。
- 8) 割込み処理のためにセーブされていたレジスタをリストアし割り込まれたプログラムに復帰する。

(5) 割込みマスクポート

8214 モードでは各割込み入力を独立に禁止する機能をコントローラがもっていないため以下に示す割込みマスクポートが用意されている。

これを使用することにより、システム内部で発生する割込みを独立にマスクすることができる。

OUT のみ E6H ビット 2～0
ビット 2：RS-232C RxREADY 割込みマスク
ビット 1：VRTC 割込みマスク
ビット 0：汎用タイマ 2 割込みマスク
OUT/IN 32H ビット 7
 サウンドコントローラ割込みのマスク

5.3 DMA 機能

VA には 4 チャンネルの優先順位付 DMA(ダイレクトメモリアクセス)機能がサポートされており、その内の 2 チャンネルはバススロットに解放されている。DMA 機能はメイン CPU に内蔵されている DMAU によって制御されている。

VA でサポートされている DMA 転送機能には以下のモードがある。

- ①リード転送 メモリ → I/O
- ②ライト転送 I/O → メモリ
- ③ペリファイ アドレス出力のみ

4 チャンネルある DMA チャンネルは以下に示すようにシステムにより割り当てられている。

チャンネル	機 能
チャンネル 0	ユーザチャンネル 1
チャンネル 1	未使用
チャンネル 2	FDC(内蔵ディスクインターフェイス)
チャンネル 3	ユーザチャンネル 2

5.4 タイマ機能

VA には以下に示すタイマがある。

- ① CPU 内 TCU 3 系統ある。
 - カウンタ#0 汎用タイマ 1 (8259 割込みモード専用)
 - カウンタ#1 BEEP 周波数設定
 - カウンタ#2 RS-232C ボーレート設定
- ② 汎用タイマ 2 (8214 割込みモード専用)
- ③ 汎用タイマ 3 (マウスタイマ)
- ④ サウンドコントローラ内タイマ 2 系統
- ⑤ FDD 用タイマ

5.4.1 汎用タイマ (TCU カウンタ#0)

汎用タイマ 1 は CPU に内蔵されている TCU 内の 1 系統で、 μ PD8253 に準拠した動作が可能である。このタイマの割込み機能は、割込みが 8259 モード時にのみ有効となり、IR0(最高優先の割込み)を発生する。

TCU カウンタ #0 は以下の動作モードで使用する。

(1) カウントモード 0 (ワンショットタイマ)

設定した周期が経過したとき 1 回だけ割込み要求を発生する。

(2) カウントモード 3 (方形波発生)

設定した周期ごとに割込み要求を発生する。

汎用タイマ 1 による割込みはカウンタの出力がローレベルからハイレベルになった時に割込み要求を出す。カウンタへの設定値と指定される周期は次の関係にある。

$$\text{周期(ns)} = \text{設定カウンタ値} \times 250.4 \text{ ns}$$

タイマの設定方法は[5.4.8 タイマコントロールユニット]で説明する。

5.4.2 BEEP 周波数発生器 (TCU カウンタ #1)

BEEP 周波数発生器は、BEEP 音の周波数を設定するためのタイマである。通常このタイマはカウントモード 3 (方形波発生)としてプログラムされ、発生する周波数が 2400Hz となるようにカウンタ値を設定する。

詳細は[5.11 サウンド機能]を参照のこと。

5.4.3 RS-232C ボーレート発生器 (TCU カウンタ #2)

このカウンタの出力パルスが RS-232C インターフェイスのコントローラである USART (μ PD8251)に供給され、ボーレートの決定に使用される。動作モードは必ずカウントモード 3 (方形波発生)で使用する。

ボーレートはこのカウンタの周期と μ PD8251 に設定するボーレートファクタとにより以下のように決定される。

ポーレート = $3993600 / \text{カウンタ設定値} / \text{ポーレートファクタ}$

詳細は[5.10 RS-232C]を参照のこと。

5.4.4 汎用タイマ2

汎用タイマ2は 88M/F シリズが内蔵している 600 Hz 固定のタイマで、割込みが 8214 モード時にのみ有効であるため主に V1/V2 モードでのみ使用される。このタイマの割込みは INT2 に入力されている。

またポート E6H により独立したマスクが可能である。

汎用タイマ2 割込みマスクポート

OUT E6H	ビット 0
0	: 割込み禁止
1	: 割込み許可

5.4.5 汎用タイマ3

汎用タイマは 15Hz・30Hz・60Hz・120Hz の4種類のインターバルで CPU に対して割込みをかけることができる独立したタイマであり、割込みモードにかかわらずに使用できる。主にマウス等の周辺機器の制御用に使用される。PC-Engine ではこのタイマは 120Hz 固定とし、マウス・フロッピーディスク・プリンタ・スクリーンエディタなどの BIOS で使用しているために自由に使用することはできない。

このタイマによる割込みレベルは、

8259 モードのとき	———	IR13 割込み
8214 モードのとき	———	INT6 割込み

となっている。

ポート 1A8H によりインターバルの指定および割込みマスクを行う。

5.4.6 サウンドコントローラ内タイマ

サウンドコントローラ (YM2203 内) が持つ FM 音源サウンドに使用するタイマ機能であり、割込みモードにかかわらず使用できる。

割込みレベルは、

8259 モードのとき	———	IR12 割込み
8214 モードのとき	———	INT4 割込み

となっている。

詳細は[5.10 サウンド機能]を参照。

5.4.7 FDD 用タイマ

FDD インターフェイス回路が持つ約 100ms のワンショットタイマであり、FDD インターフェイスを DMA モードで使用時に FDD モータ制御用に使用される。割込みモードは 8259 モード時のみ使用可能であり、レベルは IR11 (FDC からの割込み要求と同じ) に接続されている。

詳細は[5.7 ディスクインターフェイス]を参照。

5.5 カレンダ時計

5.5.1 インターフェイス方法

カレンダ時計はバッテリーによりバックアップされており、年・月・日・時・分・秒を常に計時している。4の倍数の年は閏年として自動判別される。LSIとしては μ PD1990上位互換の μ PD4990が使用されている。ただし μ PD1990互換モードにおいては年機能が使用できない。

μ PD4990とのデータインターフェイスは μ PD1990と同様にシリアル入出力により行う。コマンドインターフェイスは2種類のものがあり、VAにおいてはV1/V2モードではパラレルコマンドモード、V3モードではシリアルコマンドモードを使用する。

(1) パラレルコマンドモード (μ PD1990 互換モード)

パラレルコマンドモードでは μ PD1990と同様にすべてのコマンドを3ビットのパラレルコマンドポート(OUT 10H)から与える。この場合には μ PD1990と互換性があるモードとなるが年機能が使用できない。

パラレルコマンドとしては下記の5つのコマンドを使用する。

●パラレルコマンド：ポート 10H の下位 3 ビット

#0:	0	0	0	レジスタホールド
#1:	0	0	1	レジスタシフト
#2:	0	1	0	タイムセット&タイマホールド
#3:	0	1	1	タイムリード
#7:	1	1	1	シリアルコマンドモード設定

ポート 10H はプリンタデータと共通であり、ポート 40H で送出するストローブによって区別する。

●コマンド送出シーケンス

①ポート 10H にコマンドを出力する

```
MOV  AL, Command
OUT  10H, AL
```

②ポート 40H のビット 1 (CSTB) を 1 (ON) にする

③ポート 40H のビット 1 (CSTB) を 0 (OFF) にする

```
CLI
MOV  AL, Todctl (ポート 40H のコピー)
OR   AL, 2
OUT  40H, AL
NOP
NOP
AND  AL, 0FDH
OUT  40H, AL
MOV  Todctl, AL
STI
```

●データ送出シーケンス

下記のシーケンスをビットごとに LSB から MSB まで繰り返す

- ①ポート 10H ビット 3 にデータビットを設定する

```
MOV AL, 0 or 8
```

```
OUT 10H, AL
```

- ②ポート 40H のビット 2 (CCLK) を 1 (HIGH) にする

- ③ポート 40H のビット 2 (CCLK) を 0 (LOW) にする

```
CLI
```

```
MOV AL, Todctl (ポート 40H のコピー)
```

```
OR AL, 4
```

```
OUT 40H, AL
```

```
NOP
```

```
NOP
```

```
AND AL, 0FBH
```

```
OUT 40H, AL
```

```
MOV Todctl, AL
```

```
STI
```

●データ受取りシーケンス

下記のシーケンスをビットごとに LSB から MSB まで繰り返す

- ①ポート 40H のビット 2 (CCLK) を 1 (HIGH) にする

- ②ポート 40H のビット 2 (CCLK) を 0 (LOW) にする

```
CLI
```

```
MOV AL, Todctl (ポート 40H のコピー)
```

```
OR AL, 4
```

```
OUT 40H, AL
```

```
NOP
```

```
NOP
```

```
AND AL, 0FBH
```

```
OUT 40H, AL
```

```
MOV Todctl, AL
```

```
STI
```

- ③ポート 40H のビット 4 またはポート 1CBH ビット 0 からデータビットを読み出す。

```
NOP
```

```
NOP
```

```
IN AL, 40H
```

```
SHR AL, 4
```

```
AND AL, 1
```

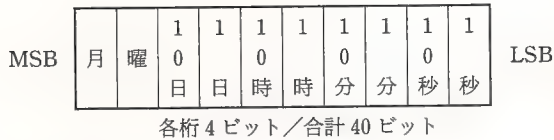
```
MOV DX, 01CBH
```

```
IN AL, DX
```

●データフォーマット

各桁のデータ形式は BCD だが、月だけは 16 進形式である。

曜日は日曜を 0・月曜を 1…とした値。



(2) シリアルコマンドモード (年機能使用モード)

シリアルコマンドは上記のパラレルコマンド #7 によりシリアルコマンドモードが設定された場合に使用でき、カレンダ時計に対するコマンドは入力データと同様にシリアルに送出される。つまりシリアルデータとして 4 ビットのコマンドを送出し、次に CSTB を送出することによってそのデータがコマンドとして取り込まれる。

注意：CSTB を送る時にはパラレルコマンドポートは 7 にしておく必要がある。

なお、シリアルコマンドには以下のものを使用する。

#0:	0	0	0	0	レジスタホールド
#1:	0	0	0	1	レジスタシフト
#2:	0	0	1	0	タイムセット&タイマホールド
#3:	0	0	1	1	タイムリード

●コマンド送出シーケンス

①パラレルコマンド #7 でシリアルコマンドモードに設定する

```

MOV    AL, 7
OUT    10H, AL
CLI
MOV    AL, Todctl (ポート 40H のコピー)
OR     AL, 2
OUT    40H, AL
NOP
NOP
AND    AL, 0FDH
OUT    40H, AL
MOV    Todctl, AL
STI

```

②下記のシーケンスをビットごとに LSB から MSB まで繰り返す。ポート 10H ビット 3 にデータビットを設定する

```

MOV    AL, 0 or 8
OUT    10H, AL

```

ポート 40H のビット 2 (CCLK) を 1 (HIGH) にする

ポート 40H のビット 2 (CCLK) を 0 (LOW) にする

```

CTI
MOV    AL, Todctl (ポート 40H のコピー)
OR     AL, 4
OUT    40H, AL
NOP

```



```

NOP
AND    AL, 0FBH
OUT    40H, AL
MOV    Todctl (ポート 40H のコピー)
STI

```

③ポート 40H のビット 1 (CSTB)を 1 (ON)にする

④ポート 40H のビット 1 (CSTB)を 0 (OFF)にする

```

MOV    AL, 7
OUT    10H, AL
CLI
MOV    AL, Todctl (ポート 40H のコピー)
OR     AL, 2
OUT    40H, AL
NOP
NOP
AND    AL, 0FDH
OUT    40H, AL
MOV    Todctl, AL
STI

```

●データ送出シーケンス (パラレルコマンド時と同じ)

下記のシーケンスをビットごとに LSB から MSB まで繰り返す

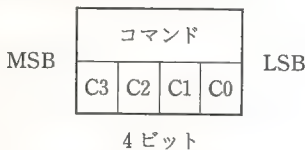
- ①ポート 10H ビット 3 にデータビットを設定する
- ②ポート 40H のビット 2 (CCLK)を 1 (HIGH)にする
- ③ポート 40H のビット 2 (CCLK)を 0 (LOW)にする

●データ受取りシーケンス (パラレルコマンド時と同じ)

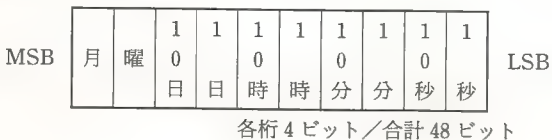
下記のシーケンスをビットごとに LSB から MSB まで繰り返す

- ①ポート 40H のビット 2 (CCLK)を 1 (HIGH)にする
- ②ポート 40H のビット 2 (CCLK)を 0 (LOW)にする
- ③ポート 40H ビット 4 またはポート 1CBH ビット 0 からデータビットを読み出す。

●コマンド/データフォーマット



各桁のデータ形式は BCD だが、月だけは 16 進形式である。
曜日は日曜を 0・月曜を 1…とした値。



5.5.2 日付／時刻の設定・読出し

設定・読出しのシーケンスはパラレルコマンド／シリアルコマンドとも同じ。

(1) 日付／時刻の設定

日付／時刻の設定は次の順序で行う。

- ①レジスタシフトコマンド(#1)を送る
- ②日付／時刻データ(40 または 48 ビット)を LSB から送る
- ③タイムセットコマンド(#2)を送る
- ④レジスタホールドコマンド(#0)を送る

(2) 日付／時刻の読出し

日付／時刻の読出しは次の順序で行う。

- ①タイムリードコマンド(#3)を送る
- ②レジスタシフトコマンド(#1)を送る
- ③日付／時刻データ(40 または 48 ビット)を LSB から受け取る
- ④レジスタホールドコマンド(#0)を送る

5.6 キーボードインターフェイス

5.6.1 キーボードインターフェイス概要

本機のキーボードには CPU 内蔵のシリアルキーボードが使用されており、キーボード上のキースイッチの操作をキーボード上のキーボードサブ CPU がスキャンし、そのデータを本体にあるインターフェイスサブ CPU に転送する。メイン CPU は、インターフェイスサブ CPU からキーの情報を受け取る。メイン CPU とインターフェイスサブ CPU とのインターフェイスには以下に示す 2 種類の機能がサポートされている。

- ①マトリクススキャン方式
- ②キーコード方式

マトリクススキャンインターフェイス機能は常時使用できるが、キーコードインターフェイスは使用できるモードに条件があり、またモードの設定が必要となる。

5.6.2 マトリクススキャン方式

(1) 方式説明

マトリクススキャン方式は 88M/F シリーズと互換性をもつための方式であり、メイン CPU はマトリクス状に定義されたキースイッチの個々のマーク／ブレイク情報を任意の時にテストすることができる。

キーボードユニット上のキースイッチが操作された場合には、キーボードサブ CPU は操作されたキーの接点情報をコードとして送信してくる。

これを受信したインターフェイスサブ CPU は、このコードをもとに定義されたキースイッチのマトリクスに従ってコードを展開し、マトリクス上の個々のキースイッチのマーク／ブレイクのデータとしてメイン CPU の入力ポート回路に入力する。

メイン CPU はこの入力ポートを任意のタイミングで読出すことができ、これにより定義され

たキーマトリクスのキースイッチの状態をテストすることができる。

またこの場合、キースイッチに必ず発生するチャタリングによる誤動作の影響はすでにキーボードサブ CPU が取り除いているため、メイン CPU はチャタリングの影響を考慮することなしにこの入力ポートでキーボードの状態をテストすることができる。

(2) I/O ポート

マトリクススキャンインターフェイスの I/O ポートは、

0000～000EH

の 15 バイトの入力ポートにより構成されている。詳細は I/O ポート一覧表のポート 0000～000EH を参照。

5.6.3 キーコード方式

(1) 方式説明

マトリクススキャンインターフェイス機能が常時使用できるのに対して、キーコード方式のインターフェイスは、割込みモードが[8259 モード]時にのみ使用することができる。また、リセット後にはこのインターフェイス機能は禁止状態となっており、このインターフェイス機能を使用する場合にはあらためてモードの設定を行う必要がある。(モードの設定は 5.6.4 を参照)

キーボード上のキーが操作された場合には、キーボードサブ CPU は操作されたキーの情報をコードとして送信してくる。これを受信したインターフェイスサブ CPU はこのコードをもとに定義されたキーに割り当てられたスキャンコードに変換してメイン CPU の入力ポートにラッチし、割込みを要求する。メイン CPU はこの割込みを処理することにより操作されたキースキャンコードを読み込むことができる。

もしもメイン CPU がこの割込み要求に応答しなければキーボードサブ CPU からの次のデータ転送が待たされることになり、その後操作されたキーのうち、キーボードユニット側のバッファをオーバーフローしたものが切り捨てられる。

(2) I/O ポート

IN 1C1H: スキャンコード

●スキャンコード表

		上位4ビット(MSBは0)→							
		0	1	2	3	4	5	6	7
下位4ビット↓	0	ESC	Q	F	<	—	.	STOP	左 SFT
	1	1!	W	G	>	/	決定	COPY	CAPS
	2	2"	E	H	/?	7		F1	カナ
	3	3#	R	J	—	8		F2	GRPH
	4	4\$	T	K	SPACE	9		F3	CTRL
	5	5%	Y	L	変換	*		F4	
	6	6&	U	;+	ROLL UP	4		F5	
	7	7'	I	:*	ROLL DOWN	5		F6	
	8	8(O]	INS	6		F7	右 SFT
	9	9)	P	Z	DEL	+		F8	RET
	A	0	@	X	↑	1		F9	PC
	B	—=	[C	←	2		F10	全角
	C	^	RETURN	V	→	3			
	D	¥	A	B	↓	=			
	E	BS	S	N	CLR	0			
	F	TAB	D	M	HELP	,			

5.6.4 コマンドインターフェイス

キーボードユニットを制御するためのコマンドポートが用意されており、キーボードシステムの動作モードを指定することができる。コマンドは1バイトであり、ポートに出力されたコマンドはインターフェイスサブ CPU に転送される。

コマンドは上位2ビットにより2種類に分けられる。

上位2ビット

1	0	オペレーションコマンド
1	1	モードコマンド

オペレーションコマンドはトリガとして実行される。例えば RESET=1 となっているオペレーションコマンドを出力すると、インターフェイス機能がリセットされるが、この後リセットを解除するために RESET=0 として改めてオペレーションコマンドを出力する必要はない。

コマンドインターフェイスはポート 197H により行われる。各ビットの意味はI/Oポート一覧表のポート 197H を参照。

注意：キーボードユニットのリセット動作

RESET を指定した場合にはインターフェイスサブ CPU の内部バッファのクリア(データバッファのクリア, モードのイニシャライズ)を行う。

インターフェイスサブ CPU はハードウェアにより入力されたりセットがパワーオンリセットかマニュアルリセットかの判断ができないため、通常のリセットシーケンスでは内部バッファのクリアを行わない。このためメイン CPU はシステムリセットが発生した場合にはそのリセット動作がパワーオンリセットかマニュアルリセットかを判断し(システムポート 5:190H による)パワーオンリセットの場合にのみ上記 RESET コマンド をインターフェイスサブ CPU に出力して内部バッファのクリアを行い、マニュアルリセット時にはインターフェイスサブ CPU をリセットしないようにする。

5.7 ディスクインターフェイス

5.7.1 FDD インターフェイス概要

本機には 2DD/2HD 兼用タイプの 5 インチフロッピーディスクドライブ(FDD)が 2 ドライブ実装されている。それを制御するインターフェイス機能には以下の 2 種類の方式があり、必要に応じ選択して使用することができる。

(1) インテリジェントモード

88M/F シリーズと互換性のあるモードで、FDD の制御を行うためにディスクサブ CPU を使用する。サブ CPU はメイン CPU とはパラレルポートで接続されてデータ/コマンドの交換が行われる。

(2) DMA モード

DMA モードでは FDD を制御するコントローラ等はすべてメイン CPU の管理下にあり直接メイン CPU により制御される。また FDD とメインメモリとの間のデータ転送は DMA により行われる。

5.7.2 動作モードの遷移

FDD インターフェイスには上記 2 種類の動作モードがサポートされているが、この 2 種類の動作モードはソフトウェアにより選択して使用することができる。ただし、DMA モードにおいては、割込み機能と DMA 機能を使用するため、それらの機能についても使用できる環境に設定する必要がある。

両モードの遷移はインテリジェントモード用のハンドシェイクポートを用いて行われる。このハンドシェイクポートはいずれのモードからもアクセスすることができ、ここを使用してサブ CPU とコマンドのハンドシェイクを行いモードの変更を行う。

システムは電源投入時/リセット時にはインテリジェントモードとなっており、そのままの状態では 88M/F シリーズと互換性のあるインターフェイスとして使用することができる。

5.7.3 I/O ポート

モード設定/インテリジェントモード制御を行う場合に操作する必要がある I/O ポートは以下のとおり。

IN	FCH	データ入力ポート
OUT	FDH	コマンド／データ出力ポート
OUT	FEH	ハンドシェイク制御ポート
OUT	1B0H	インターフェイスモード設定ポート
	ビット 0	: MODE インターフェイスモード
	0	: インテリジェントモード
	1	: DMA モード

5.7.4 インテリジェントモード

(1) 概要

インテリジェントモードでは FDD の制御はすべてディスクサブシステムの CPU (μ PD780) が行い、メイン CPU はハンドシェイクポートを介してコマンド／データの転送を行う。このインターフェイスモードの機能については、以下の点を除いて 88M シリーズと互換性がある。

- ① ハンドシェイクポート用パラレルポートは μ PD8255 のサブセットとなっており、モード 0 の機能およびポート C のビットセット／リセット機能のみがサポートされている。それ以外の μ PD8255 の機能はサポートされていない。
- ② サブシステム側の FDD 制御インターフェイスには、アナログ VFO 回路が使用されているため、デジタル VFO 制御用のコマンドは削除されている。
- ③ インターフェイスモードの遷移を行うためのスリープコマンド、ウェイクアップコマンドが新たにサポートされている。

(2) ハンドシェイクポート

① データ入力ポート

IN FCH FD サブシステムからのデータ入力ポート

② データ出力ポート

OUT FDH FD サブシステムへのデータ出力ポート

③ ハンドシェイク制御ポート

IN/OUT FEH ハンドシェイクの制御ポート

④ ポート制御ポート

OUT FFH インターフェイスの初期化
ハンドシェイク制御ポートの操作

(3) FDD サブシステム

FDD サブシステムの仕様を以下に示す。

- | | | | |
|--------|--------------|--------------|--------|
| ① CPU | μ PD780A | 4MHz | ウェイト無し |
| ② ROM | 8K バイト | (0000~1FFFH) | |
| ③ RAM | 16K バイト | (4000~7FFFH) | |
| ④ FDC | μ PD765A | アナログ VFO | 搭載 |
| ⑤ PORT | μ PD8255 | サブセット | |

FDD サブシステムの I/O ポートは以下の通り。

●割込みオペコードポート

OUT F0H

割込み発生時に CPU がフェッチする命令コード

リセット時／通常時 7FH

●ドライブ制御ポート

	7	6	5	4	3	2	1	0
OUT F4H	—	—	CLK	DSI	TD1	TD0	RV1	RV0

- RV0／RV1 : ドライブ 0／1 のモード
0 : 2D／2DD モード
1 : 2HD モード
TD0／TD1 : ドライブ 0／1 のトラック密度
0 : 48 TPI (2D)
1 : 96 TPI (2DD／2HD)
DSI : FDC のドライブセレクト禁止
0 : 許可
1 : 禁止
CLK : FDC の動作クロックの選択
0 : 4.8MHz(2D／2DD)
1 : 8 MHz(2HD)

リセット時

7	6	5	4	3	2	1	0
—	—	0	0	0	0	0	0

(注) ドライブのトラック密度を 48tpi から 96tpi に変更する場合には、まず DSI=1 にした後切換えを行う必要がある。

●モータ制御ポート

	7	6	5	4	3	2	1	0
OUT F8H	—	—	—	—	PCM	—	M1	M0

- M0／M1 : ドライブ 0／1 のモータ制御
0 : モータ OFF
1 : モータ ON
PCM : プリコンペンセーション制御
0 : プリコンペンセーション OFF
1 : プリコンペンセーション ON

リセット時

7	6	5	4	3	2	1	0
—	—	—	—	0	—	0	0

(注)モータ ON の際は、動作可能となるまでに 600 ms の待ち時間を必要とする。

●ターミナルカウントポート

入力ポート： F8H 入力データに意味はない。

機能 : FDC にターミナルカウント(TC)パルスを出力する。

● FDC ステータスレジスタ

	7	6	5	4	3	2	1	0
OUT FAH	RQM	DIO	NDM	CB	—	—	D1B	D0B

D0B/D1B : ドライブ 0/1 のビジィ信号
 0 : レディ
 1 : ビジィ
 CB : FDC の ビジィ信号
 0 : レディ
 1 : ビジィ
 NDM : FDC の NON—DMA 信号
 0 : OFF
 1 : ON
 DIO : FDC の データ I/O 信号
 0 : CPU → FDC
 1 : FDC → CPU
 RQM : FDC の マスタ要求信号
 0 : DIO=0 CPU のデータセット完了
 DIO=1 CPU のデータ引取り完了
 1 : DIO=0 FDC のデータ引取り完了
 DIO=1 FDC のデータセット完了

● FDC データレジスタ

	7	6	5	4	3	2	1	0
I/O FBH	コマンド／データ／リザルトステータス 等							

● ハンドシェイクポート

・ データ入出力ポート

IN FCH メインシステムからのデータ入力ポート
 OUT FDH メインシステムへのデータ出力ポート

・ ハンドシェイク制御ポート IN/OUT FEH

	7	6	5	4	3	2	1	0
I/O FEH	—	DACO	RFDO	DAVO	ATNI	DACI	RFDI	DAVI

DACO : メインシステムに対する DAC 出力
 0 : ビジィ
 1 : サブシステムがデータ引取り完了
 RFDO : メインシステムに対する RFD 出力
 0 : ビジィ
 1 : サブシステムハンドシェイクレディ
 DAVO : メインシステムに対する DAV 出力
 0 : サブシステムはデータを出力していない
 1 : サブシステムがデータを出力中
 ATNI : メインシステムからの ATN 入力
 0 : メインシステムはコマンドを出力していない
 1 : メインシステムがコマンドを出力中
 DACI : メインシステムからの DAC 入力
 0 : ビジィ
 1 : メインシステムがデータ引取り完了

RFDI	: メインシステムからの RFD 入力
0	: ビジィ
1	: メインシステムハンドシェイクレディ
DAVI	: メインシステムからの DAV 入力
0	: メインシステムはデータを出力していない
1	: メインシステムがデータを出力中

・モード設定／ビット操作 OUT FFH

モード設定

OUT FFH, 91H

ビット操作

ポート FEH の出力ビットを操作する。

OUT FFH

ビット 7 0

ビット 3～1 100=DAVO

101=RFDO

110=DACO

ビット 0 1=セット／0=リセット

5.7.5 DMA モード

(1) 概要

DMA モードでは FDD の制御はすべてメイン CPU が行い、FDC(μ PD765A)とメインメモリ間のデータ転送には DMA 転送が使用される。

DMA モードで FDD の制御を行うためには、FDD インターフェイスモードとして DMA モードを選択するだけでなく、以下の条件が必要となる。

- ① 割込みモードが 8259 モードであること。
- ② DMA チャンネル 2 を使用すること。
- ③ VFO 回路にはアナログ VFO 回路が使用されている。

フロッピーディスクにリード／ライトするためのデータを、メインメモリと FDC との間で DMA 転送する際に使用される DMA チャンネルはチャンネル 2 である。

DMA モードにおいてメイン CPU が FDC の制御を行う場合、割込み制御機能が必要となるが、FDC からの割込み入力の処理は割込みモードが[8259 モード]の場合のみしか行えないため、割込みモードとして[8259 モード]を設定する必要がある。この場合使用される割込みチャンネルは IR11 である。

(2) I/O ポート

ビットアサインは I/O ポート一覧表を参照。

FDC 制御ポート 0 (1B2H)

ドライブ別モード設定(2DD／2HD)

ドライブ別トラック密度(48／96TPI)

FDC のドライブセレクト禁止／許可

FDC の動作クロックの選択(4.8MHz／8MHz)

- FDC 制御ポート 1 (1B4H)
 - ドライブ別モータ制御
 - プリコンペンセーション制御
- FDC 制御ポート 2 (1B6H)
 - FDC リセット
 - FDC 強制レディ
 - DMA 制御
 - FDC タイマ制御
- FDC(μ PD765A)レジスタ (1B8H/1BAH)
 - FDC ステータス
 - コマンド/データ/リザルトステータス

● FDC 強制レディの動作

FDC 強制レディ機能は、ドライブの有無をチェックするために行うリキャリブレイトコマンドだが、ドライブが接続されていなくても実行が開始されるように、FDC(μ PD765A)に対して強制的にドライブレディ信号を入力するためのもので、このリキャリブレイト動作が正常終了したドライブのみが接続されていると判断される。

● FDC タイマ機能

FDD インターフェイスには独立したリトリガブルな約 100 ms のワンショットタイマ機能があり必要に応じて使用することができる。

このタイマのカウント終了は、割込み要求として出力される。ただし、この割込み要求は FDC からの割込み要求と OR されて IR11 に出力されているため、FDC が動作中には使用できない。このタイマ機能は主にタイムアウトによる FDD のモータ OFF 機能を実現するために使用される。

また、このタイマを使用していない場合には必ず割込みマスクをしておく。

FDC タイマの制御はポート 1B6H のビット 2 (FDC タイママスク)とビット 0 (タイマトリガ)とによって行われる。FDC タイマ制御シーケンスは以下の通り。

- ①まず FDC タイマにトリガをかける。
- ② FDC タイマのマスクを解除する。
- ③必要に応じて再びトリガをかけることができる。

割込み要求が発生する以前(100 ms 以内)に再トリガを行うと、再トリガのタイミングから再びカウントが開始される。

- ④トリガした時点(①または③)より約 100 ms 後に割込みが発生する。
- ⑤割込みを処理する。
- ⑥ FDC タイマにマスクをかける。

5.7.5 HD インターフェイス

HD インターフェイスは PC-9800 シリーズと共通であり、PC-9800 シリーズ用 HD ユニットが使用できる。

5.8 プリンタインターフェイス

5.8.1 プリンタインターフェイス概要

プリンタインターフェイスは、セントロニクス社仕様に準拠した8ビットパラレルインターフェイスとなっている。またストローブはソフトウェアで送出する。

5.8.2 I/Oポート

以下にインターフェイスが使用する I/O ポートを示す。

(1) データポート

OUT 10H : 出力データ

(2) ストローブ ポート

OUT 40H (システムポート3)

ビット0	: プリントストローブ
0	: ストローブ ON
1	: ストローブ OFF

(3) ステータスポート

IN 40H (システムポート4)

ビット0	: プリント ビジィ
0	: プリント レディ
1	: プリント ビジィ

5.9 RS-232C インターフェイス

5.9.1 RS-232C インターフェイス概要

RS-232C インターフェイスは、EIA で規格化されているインターフェイスで、モデム等との接続に使用する。VA では、割込み制御の RS-232C インターフェイスを1チャンネル内蔵しており汎用のシリアルインターフェイスとして使用できる。RS-232C インターフェイスには μ PD8251AFC が使用されている。

5.9.2 I/Oポート

(1) μ PD8251 制御ポート

OUT/IN 20H	データポート
OUT/IN 21H	モード/コマンド/ステータス

(2) 割込みマスクポート

割込みマスクポートは、設定された割込みモードによりその I/O アドレスが異なる。割込みモード及び、割込み制御の方法については[5.2 割込み]で説明されている。

① 8214 モードの場合

OUT E6H

ビット 2 : RS-232C RxRDY 割込みマスク

0 : 禁止

1 : 許可

② 8259 モードの場合

OUT 1CDH (システムポート 8)

ビット 2 : RS-232C TxRDY 割込みマスク

ビット 1 : RS-232C TxEMPTY 割込みマスク

ビット 0 : RS-232C RxRDY 割込みマスク

0 : 禁止

1 : 許可

(注)ポート 1CFH により各ビット単位にセット／リセットを行うことができる。

(3) ステータスポート : (システムポート 4, 7)

① DCD 信号

IN 40H ビット 2

IN 1CBH ビット 5

0 : キャリア OFF

1 : キャリア ON

② CTS 信号

IN 1CBH ビット 6

0 : CTS OFF

1 : CTS ON

③ CI 信号

IN 1CBH ビット 7 CI 信号

0 : CI OFF

1 : CI ON

(4) ボーレート制御ポート

RS-232C のボーレートは CPU に内蔵されている TCU のカウンタ #2 をプログラムすることにより決定される。

OUT/IN 1A4H カウンタ #2 データステータス

OUT/IN 1A6H TCU 制御(10110110B)

タイマのカウントレートは 250.4 ns(3.9936 MHz)となっており、使用するボーレートの設定は μ PD8251 に設定するボーレートファクタと関連してカウンタ #2 にカウント数を設定することにより行われる。

ボーレート	同期 X1	非同期 X16	非同期 X64
19200	208	13	使用不可
9600	416	26	使用不可
4800	832	52	13
2400	1664	104	26
1200	3328	208	52
600	6656	416	104
300	13312	832	208
150	26624	1664	416
75	53248	3328	832

5.10 サウンド機能

5.10.1 サウンド機能概要

本機には FM 音源方式のデジタルシンセサイザが搭載されており、自然楽器に近い音を合成して出力することができる。

サウンド機能としては以下のものがある。

- 1) FM 音源 : 3色 3音 (4 オペレータ)
- 2) SSG 音源 : 3音
- 3) BEEP : タイマで設定された周波数の音を出す
- 4) PORT : 1 ビットの出力ポートで直接スピーカーをドライブ

5.10.2 サウンドコントローラ

FM 音源／SSG 音源には専用 LSI(YM2203)が使用されている。YM2203 の詳細については、YM2203 のドキュメントを参照するものとして、ここでは I/O アドレスとレジスタの機能について記述するにとどめる。

(1) YM2203 の I/O アドレス

OUT 44H	レジスタの選択
IN 44H	ステータスの読出し
OUT 45H	レジスタへの書込み
IN 44H	レジスタから読出し

また 46H, 47H はサウンド機能の将来の拡張のためにリザーブされている。

(2) YM2203 のレジスタ構成

YM2203 の内部レジスタのアクセス方法は間接アドレス方式となっており、まずアクセスしたいレジスタを指定した後、データポートよりアクセスする。

なお、①レジスタを指定した後データを入出力するまでと、②データ入出力後次のレジスタを指定するまでには YM2203 がビジーとなる期間があるため、レジスタの設定／アクセスはステータス内にあるビジーフラグ(OPNBSY 信号)がレディとなっていることを確認した後行う必要がある。

(2.1) YM2203 の内部レジスタの構成

アドレス 00～0FH まではリード／ライト可能

アドレス	レジスタ						コメント		
00H 01H 02H 03H 04H 05H	チャンネル A 周波数設定 (下位 8) (上位 4) チャンネル B 周波数設定 (下位 8) (上位 4) チャンネル C 周波数設定 (下位 8) (上位 4)						SSG		
06H	ノイズジェネレータ制御						SSG		
07H	I/O 制御		ミキサー制御				SSG, I/O		
08H 09H 0AH	チャンネル A レベル チャンネル B レベル チャンネル C レベル						SSG		
0BH 0CH	エンベロープ パラメータ (下位 8) (上位 8)						SSG		
0DH	—	—	—	—	C	ATT	ALT	HLD	SSG
0EH 0FH	入力ポート 入出力ポート						I/O		

アドレス 24H～はライトのみ

アドレス	レジスタ		コメント
24H 25H 26H	タイマ A (上位 8) (下位 2) タイマ B		タイマ
27H	モード	タイマ制御	タイマ, FM
28H	各チャネルに対するスロットの ON/OFF		FM
2DH 2EH 2FH	プリスケラ制御		FM
30H 38H 34H 3CH 31H 39H 35H 3DH 32H 3AH 36H 3EH	チャンネル 1 : スロット 1 スロット 2 スロット 3 スロット 4 チャンネル 2 : スロット 1 DETUNE スロット 2 /MULTIPLE スロット 3 スロット 4 チャンネル 3 : スロット 1 スロット 2 スロット 3 スロット 4		FM
40H } 4EH	同上 トータルレベル (アドレスの順序に注意)		FM

50H └ 5EH	同上	KEY スケール ATTACK レート	FM
60H └ 6EH	同上	DECAY レート	FM
70H └ 7EH	同上	SUSTAIN レート	FM
80H └ 8EH	同上	SUSTAIN レベル RELEASE レート	FM
90H └ 9EH	同上	SSG タイプの エンベロープ 制御	FM
A0H A4H A1H A5H A2H A6H	チャンネル 1 F-Num. (上位), BLOCK チャンネル 2 F-Num. (下位) F-Num. (上位), BLOCK チャンネル 3 F-Num. (下位) F-Num. (上位), BLOCK		FM 通常モード * 1 * 1
A9H ADH AAH AEH A8H ACH	スロット 1 F-Num. (上位), BLOCK スロット 2 F-Num. (下位) F-Num. (上位), BLOCK スロット 3 F-Num. (下位) F-Num. (上位), BLOCK		FM チャンネル 3 通常モード 以外
B0H B1H B2H	チャンネル 1 チャンネル 2 チャンネル 3	SELF FEEDBACK / CONNECTION	FM

* 1 通常モード以外の場合には、チャンネル 3 のスロット 4 の指定となる。

(2.2) ステータスレジスタフォーマット

7						1	0
OPNBSY	—	—	—	—	—	TFLGB	TFLGA

OPNBSY: コマンド/データインターフェイスビジィフラグ
0 = レディ / 1 = ビジィ
TFLGB: タイマ B 終了フラグ
0 = セット状態 (カウント終了)
1 = リセット状態
TFLGA: タイマ A 終了フラグ
0 = セット状態 (カウント終了)
1 = リセット状態

(3) YM2203 からのタイマ割り込み

YM2203 には 2 系統のタイマ機能があり、このタイマにより CPU に割り込みをかけることができる。割り込みのレベルはモードによって異なる。

- 1) 8214 モード ——— INT 4 割込み
- 2) 8259 モード ——— IR 12 割込み

なお、8214 モードの場合には YM2203 からの割込み出力はバススロットからの割込み入力 UINT1(IR5)との論理和をとって割込みコントローラに入力されている。したがって YM2203 のタイマ割込み機能を使用する場合には UINT1(IR5)からの割込み入力を使用することができない。

YM2203 からの割込みには独立したマスク機能があり、ポート 32H により制御できる。

OUT 32H

ビット 7	: SINTM サウンド割込みマスク
0	: 割込み許可
1	: 割込み禁止

(例: サウンドコントローラ割込みを禁止する場合)

```

GMode8 EQU 32H
        CLI
        IN    AL, _GMode8
        OR    AL, 80H
        OUT   _GMode8, AL
        STI

```

この割込みマスクは YM2203 内からの割込みを使用しない場合だけでなく、YM2203 内にある 2 組のタイマを同時に使用する時にも使用される。

この場合 YM2203 からの割込みに応答する割込み処理ルーチンでは、割込みを受け付けいたらまず SINTM = 1 にして割込み信号をマスクした後で割込み処理を行い、割込み要因を取り除いた後(カウントアップしたタイマをリセットする)に SINTM = 0 としてマスクを解除する。この操作により割込み機能がロックすることが防止される。

5.10.3 BEEP

BEEP 回路は TCU のカウンタ#1 で設定した周波数でスピーカーを鳴らすもので、ポート出力によって ON/OFF する。これはシステムで使われるブザー音として使用されている。

発生する音の周波数は通常 2400Hz に設定されているが、カウンタ#1 の設定値を変更することによって周波数を変更することができる。

周波数(Hz) = 3993600 / カウンタ設定値

BEEP の ON/OFF を指定するポートには以下に示す 2 ビットがあり、そのいずれか一方が ON となると他方が OFF でも BEEP は ON となるが、V3 モードではポート 1CDH を使用し、ポート 40H の BEEP は常に 0 にしておく。

OUT 40H

ビット 5	: BEEP ブザー ON/OFF 指定
0	: ブザー OFF
1	: ブザー ON

OUT 1CDH

ビット 3 : XBEEP ブザー ON/OFF 指定
 0 : ブザー ON
 1 : ブザー OFF

(例)

ポート 1CDH の操作はポート 1CFH によるビットセット/リセットにより行う。

```
_SysPort EQU 1CFH
MOV DX, _SysPort
MOV AL, 6
OUT DX, AL ; BEEP ON
:
MOV DX, _SysPort
MOV AL, 7
OUT DX, AL ; BEEP OFF
```

5.10.4 PORT

ポート出力による音声出力は1ビットの出力ポートが直接スピーカーをドライブするもので、ソフトウェアにより周波数が決定される。またこの機能には独立したマスク機能があり、通常はマスクされている。

● PORT 出力

OUT 40H(システムポート 3)

ビット 7 (FBEEP)の連続的な ON/OFF によってサウンドを発生させる。

● PORT マスク

OUT 190Hビット(システムポート 5)

ビット 4 : FBEEP イネーブル(FBEN)
 0 : FBEEP によるコントロールを禁止
 1 : FBEEP によるコントロールを許可

5.11 マウスインターフェイス

5.11.1 マウスインターフェイス概要

マウスインターフェイスは 88SR 以降に装備されている汎用入出力インターフェイスと同等であり、マウスだけでなくジョイスティック・タブレット等各種の周辺機器を接続することができる。マウスインターフェイスポートのビット数は次の通り。

入力ポート	4 ビット
出力ポート	1 ビット
入出力ポート	2 ビット

●入力ポートからのデータ入力

入力ポートからは YM2203 を介してデータを受け取る。

OUT 0044H, 0EH (YM2203 レジスタ選択)
 IN 0045H
 ビット 0～3 : 入力データ (正論理)

●出力ポートへのデータ出力

OUT 0040H
 ビット 6 : 出力データ (正論理)

●入出力ポートの方向設定

入出力ポートの方向は 2 ビットとも一括して設定される。

OUT 0044H, 07H (YM2203 レジスタ選択)
 OUT 0045H
 ビット 7 : 入出力ポートの方向設定
 0 : 入力ポートに設定
 1 : 出力ポートに設定

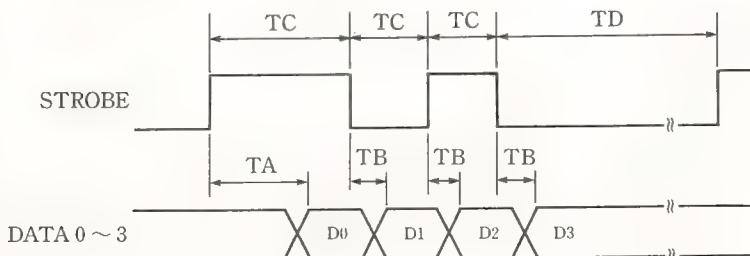
●入出力ポートでのデータ入出力

OUT 0044H, 0FH (YM2203 レジスタ選択)
 IN/OUT 0045H
 ビット 0 / 1 : 入出力データ (正論理)

5.11.2 マウスとのインターフェイス方法

マウスインターフェイスにマウスを接続する場合は各ポートを次の用途に使用している。

- ・入出力ポート (2 ビット)
 入力ポートに設定してマウスボタンの状態を取り込む
- ・入力ポート (4 ビット)
 マウスの X/Y 方向の移動量を取り込む
- ・出力ポート (1 ビット)
 マウス移動量の取込み時にストローブを送出する
- ・マウスインターフェイスタイミング
 マウスとのインターフェイスのタイミングは下図のとおり。



TA/TB はマウスが必要とする時間であり、TC/TD は CPU が守る必要がある時間。

TA : ストロープの立ち上がりから最初のデータが確定するまで=80 μ s 以内

TB : ストロープの変化から次のデータが確定するまで=30 μ s 以内

TC : データ読取り時のストロープの間隔=180us 以内

TD : 最後のストロープ変化から次のデータ読取りのためのストロープの立ち上がりまで=88 μ s 以上

D0-D3: マウスからのデータ(4ビットづつ)

前回からの移動量が符号付き1バイト数で得られる。

D0: 前回からの水平移動量(上位4ビット)

D1: 前回からの水平移動量(下位4ビット)

D2: 前回からの垂直移動量(上位4ビット)

D3: 前回からの垂直移動量(下位4ビット)

5.12 スキャナインターフェイス

5.12.1 スキャナインターフェイス概要

スキャナインターフェイスはイメージスキャナを接続するためのものであり、3個の8ビットパラレルポートから構成されている。

このI/Oポートには μ PD8255が使用されており、リセット後に各ポートの設定仕様に合わせてコマンドポートよりイニシャライズを行う必要がある。なお、イニシャライズ後には、出力ポートとして設定されたポートの出力はすべて1度[0]となるため注意が必要である。

5.12.2 I/Oポート

BCH μ PD8255 ポート A

BDH μ PD8255 ポート B

BEH μ PD8255 ポート C

BFH μ PD8255 制御ポート

スキャナを接続した場合のポートアサインをしめす。ポートCに対する出力はポートBFHを使って行う(⑤参照)。

① μ PD8255 イニシャライズ

グループA ——— モード1 ポートA出力 ポートC上位出力

グループB ——— モード0 ポートB入力 ポートC下位入力

OUT 00BFH, 0A3H

② スキャナ制御(IN/OUT)

ポートC ビット0: スキャナの接続入力

0 : スキャナが接続されている

1 : スキャナが接続されていない

ポートC ビット4: スキャナリセット出力

0 : リセット出力

1 : 動作中

ポート C ビット 1 : XSRDY レディ入力

0 : スキャナレディ

1 : スキャナビジィ

③スキャナに対する出力

ポート A : 出力データ (8 ビット)

ポート C

ビット 7 : XSOBF 読み要求出力

ポート A に出力したとき自動的に ON (0) になる。

スキャナからのデータ受信アクノリッジによって自動的に OFF (1) になる。

ビット 6 : XSACK スキャナデータアクノリッジ入力

スキャナがデータを受け取ったとき ON (0) のストロブが返される。

ビット 3 : SINTR 使用しない

④スキャナからの入力

ポート B : 入力データ (8 ビット)

ポート C

ビット 5 : XCACK データアクノリッジ出力

スキャナからのデータを受け取ったとき ON (0) のストロブを送出する。

ビット 2 : XSDIRQ 読み要求入力

スキャナがデータ出力したとき ON (0) になる。

アクノリッジを出力することにより OFF (1) になる。

⑤ポート C ビットセット/リセット

OUT BFH

ビット 7 : 0

ビット 3 ~ 1 : 操作ビット指定

ビット 0 : セット (1) リセット (0) 指定

第6章 BIOS

6.1 フロッピーディスク BIOS

内蔵フロッピーディスクドライブを DMA インターフェイスから制御するものであり、以下の機能を持っている。

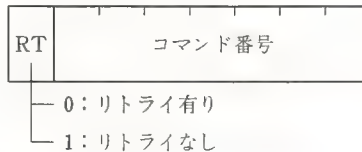
●フロッピーディスク BIOS 機能一覧

機能番号 (16 進)	機 能
00	フロッピーディスクシステムの初期化
01	データの読出し
02	データの書込み
03	1トラックフォーマット
04	1ドライブフォーマット
05	セクタ ID の読出し
06	リキャリプレート
07	実行結果ステータスの取得
08	ドライブのレディチェック
09	ドライブドアの開閉取得
0A	ディスクモードの設定
0B	ワークエリアの初期化

●呼出し時のパラメータ

各ファンクションを呼び出す時のパラメータの一般型は次の通りである。

AH = コマンド番号

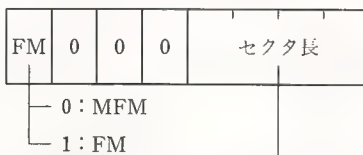


CH = ドライブ番号 (0 または 1)

CL = トラック番号 (= シリンダ番号 × 2 + ヘッド番号)

DH = セクタ番号

DL = セクタ長および MFM/FM 指定



指定値	0	1	2	3	4	5	6	7	8
バイト長	128	256	512	1024	2048	4096	8190	16384	32768

ES : BP = リード/ライト時のデータバッファ

(注) 奇数番地でもよい。FFFFFH をまたぐ領域は設定できない。

AL = データ長/フォーマットデータ/ディスクモードなど

●呼出し方

- ① パラメータ設定
- ② AH ← 機能番号
- ③ INT 80H

レジスタは AX/CF 以外は値を返す場合を除いて保存される。

●ステータス

各ファンクションは終了時に AH レジスタにリザルトステータスを返す。エラー状態のときは同時にキャリーフラグをセットする。

CF=0 (ノーエラー)

AH の値(16 進)

- 00 正常終了
- 01 ライトプロテクト(SENSE 実行時のみ)

CF=1 (エラー)

AH の値(16 進)

- 02 パラメータエラー
指定したパラメータがレンジ外の時。
バッファエリアが FFFFFH をまたぐとき。
- 03 デバイス異常
ドライブの異常/ドライブが接続されていないなどの場合。
- 04 ノットレディ
ドライブがレディ状態にない場合。
- 05 ライトプロテクト
書き込み時にライトプロテクト状態が検出されたとき。
- 06 CRC エラー(ID)
ID の CRC エラーが検出されたとき。
- 07 CRC エラー(DATA)
DATA 部の CRC エラーが検出されたとき。
- 08 ノーセクタ
トラックに指定した ID が見つからないとき。
- 09 ノーシリンダ
エラー 08 の特殊な場合で、指定した ID のシリンダ番号が見つからないとき、シークエラーの可能性があり、RECALIBRATE の必要がある。
- 0A 不良シリンダ
エラー 09 の特殊な場合で、シリンダ番号が FFH のとき。
- 0B ミッシングアドレスマーク(ID)
トラックに正常な ID がひとつも見つからないとき。
- 0C ミッシングアドレスマーク(DATA)
ID のあとに正常な DATA 部が続かないとき。
- 0D タイムアウトエラー
- FF 不定エラー(その他のエラー)
例えば、リードアフターライト時に FDC エラーが検出されないにもかかわらず、データに差異が生じた場合。

6.1.1 ファンクション解説

00H	フロッピーディスクシステムの初期化
-----	-------------------

入 力 AH=00H
INT 80H

出 力 AH=0
CF=0

(作業領域)

BYTE[DISKMODE # 0, # 1]	← FFH	(全無効モード)
BYTE[SURFMODE]	← 03H	(全両面モード)
BYTE[BIOSMODE]	← 0	(通常アクセス)
BYTE[RETRYNUM]	← 8	(リトライ 8 回)
BYTE[MTOFFTIM]	← 250D	(25 秒)
BYTE[UNITSTAT]	←	ドライブの接続状況
BYTE[DINTSTAT]	← 0	(割込み状況初期化)
BYTE[DOORSTAT]	← 0	(ドア開閉無)
BYTE[MOTRSTAT]	← FFH	(コマンド終了直後)
BYTE[SINTSTAT # 0, # 1]	← FDC の ST0	(RICALB 時)
BYTE[SINTSTAT # 0, # 1+1]	← FDC の PCN	(RICALB 時)
BYTE[MTOFFCNT]	← 250D	(コマンド終了直後)

解 説 フロッピーディスク BIOS の初期化を行う。

- 1) FDC (μPD765A) の初期化
FDC の SENSE INT STATUS コマンドを用いて、状態遷移が検出されなくなるまで ST0 を受取り続ける。
- 2) FDC SPECIFICATION の初期化
FDC の SPECIFY コマンドを用いて、SRT, HUT, HLT および ND を 1D/2D 状態に初期化する。

SPECIFY	SRT, HUT	HLT, ND
03H	DAH	18H

- 3) ヘッド位置の初期化
ドライブ #0, #1 に対して FDC の コマンドを発行し、さらに SENSE INT STATUS コマンドを用いて、各ドライブの接続状況を[UNITSTAT]に格納する。
- 4) BIOS のワークエリアの初期化
- 5) ドライブ制御ポートの初期化
ポート 01B2H を 1D/2D 状態に初期化する

01H	データの読み出し
-----	----------

入 力 AH=01H/81H

AL=セクタ数

CH=論理ドライブ番号 (US1, US0)

CL=論理トラック番号 (NCN, HD)

DH=開始セクタ番号 (R)

DL=FM 指定およびセクタ長 (N, MF)

ES:BP=バッファ領域のアドレス

BYTE[BIOSMODE]の IDR=1 の時:

BH=ID シリンダ番号 (C)

BL=ID ヘッド番号 (H)

INT 80H

(作業領域)

BYTE[SURFMODE] =サーフェスモード

BYTE[DISKMODE #0/#1] =ドライブのディスクモード

BYTE[RETRYNUM] =リトライカウント

BYTE[BIOSMODE]の PARM ビット =ユーザパラメータ使用の有無

出 力 AH=ステータスコード

CF=エラーフラグ

(作業領域)

BYTE[RSLTSTAT] =FDC のリードデータコマンド

BYTE[RSLTSTAT+1~+7] =FDC の ST0~ST2, C, H, R, N

BYTE[SINTSTAT #0/#1] =FDC の ST0(シーク時)

BYTE[SINTSTAT #0/#1+1] =FDC の PCN(シーク時)

解 説 ディスクからデータを読み出す。ただし、2トラック以上にまたがってアクセスすることはできない。[BIOSMODE]の IDR ビットに 1 が立っている場合には、レジスタ BH/BL が ID のシリンダ/ヘッド番号を指定する。[BIOSMODE]の PARM ビットに 1 が立っている場合には、FDC 用のパラメータとして[USERPARM]の 5 バイトの値を用いる。したがって、この場合にはあらかじめ[USERPARM]に値が設定されている必要がある。

02H	データの書き込み
-----	----------

入 力 AH=02H/82H

AL=セクタ数

CH=論理ドライブ番号 (US1, US0)

CL=論理トラック番号 (NCN, HD)

DH=開始セクタ番号 (R)

DL=FM 指定およびセクタ長 (N, MF)

ES:BP=バッファ領域のアドレス

BYTE[BIOSMODE]のIDR=1の時：

BH=ID シリンダ番号 (C)

BL=ID ヘッド番号 (H)

INT 80H

(作業領域)

BYTE[SURFMODE]	=サーフェスモード
BYTE[DISKMODE # 0/# 1]	=ドライブ # 0/# 1 のディスクモード
BYTE[RETRYNUM]	=リトライカウント
BYTE[BIOSMODE] : PARM	=ユーザパラメータ使用の有無
BYTE[BIOSMODE] : RAW	=チェックリードの有無

出 力 AH=ステータスコード

CF=エラーフラグ

(作業領域)

BYTE[RSLTSTAT]	=FDC のライトデータコマンド
BYTE[RSLTSTAT+1~+7]	=FDC の ST0~ST2, C, H, R, N
BYTE[SINTSTAT # 0/# 1]	=FDC の ST0(シーク時)
BYTE[SINTSTAT # 0/# 1+1]	=FDC の PCN(シーク時)

解 説 ディスクにデータを書き込む。2トラック以上にまたがってアクセスすることはできない。

03H	1トラックフォーマット
-----	-------------

入 力 AH=03H/83H

AL=イニシャライズデータ

CH =論理ドライブ番号 (US1, US0)

CL =論理トラック番号 (NCN, HD)

DL =FM 指定およびセクタ長(N, MF)

BYTE[BIOSMODE]のIDR=1の時：

ES : BP=ID 並びのアドレス

INT 80H

(作業領域)

BYTE[SURFMODE]	=サーフェスモード
BYTE[DISKMODE # 0/# 1]	=ドライブ # 0/# 1 のディスクモード
BYTE[RETRYNUM]	=リトライカウント
BYTE[BIOSMODE] : PARM	=ユーザパラメータ使用の有無

出 力 AH=ステータスコード

CF=エラーフラグ

(作業領域)

BYTE[RSLTSTAT]	=FDC のライト ID コマンド
BYTE[RSLTSTAT+1~+7]	=FDC の ST0~ST2, C, H, R, N

BYTE[SINTSTAT#0/#1] =FDC の ST0(シーク時)
 BYTE[SINTSTAT#0/#1+1] =FDC の PCN(シーク時)

解 説 指定されたトラックを、フォーマットする。[BIOSMODE]の IDR ビットに 1 が立っている場合には、レジスタ ES:BP で ID 並びの先頭アドレスを指定する。[BIOSMODE]の PARM ビットに 1 が立っている場合には、FDC 用のパラメータとして[USERPARAM]の 5 バイトの値を用いる。

04H	1 ドライブフォーマット
-----	--------------

入 力 AH=04H/84H

AL=イニシャライズデータ

CH=論理ドライブ番号 (US1, US0)

DL=FM 指定およびセクタ長 (N, MF)

BYTE[BIOSMODE]:IDR=1 の時

ES:BP=ID 並びのアドレス

INT 80H

(作業領域)

BYTE[SURFMODE] =サーフェスモード

BYTE[DISKMODE#0/#1] =ドライブ #0/#1 のディスクモード

BYTE[RETRYNUM] =リトライカウント

BYTE[BIOSMODE]:PARM =ユーザパラメータ使用の有無

出 力 AH=ステータスコード

CF=エラーフラグ

(作業領域)

BYTE[RSLTSTAT] =FDC のライト ID コマンド

BYTE[RSLTSTAT+1~+7] =FDC の ST0~ST2, C, H, R, N

BYTE[SINTSTAT#0/#1] =FDC の ST0(最終シーク時)

BYTE[SINTSTAT#0/#1+1] =FDC の PCN(最終シーク時)

解 説 指定されたドライブのメディアでフォーマットする。[BIOSMODE]の IDR ビットに 1 が立っている場合には、レジスタ ES:BP で ID 並びの先頭アドレスを指定する。[BIOSMODE]の PARM ビットに 1 が立っている場合には、FDC 用のパラメータとして[USERPARAM]の 5 バイトの値を用いる。

05H	セクタ ID の読出し
-----	-------------

入 力 AH=05H/85H

CH=論理ドライブ番号 (US1, US0)

CL=論理トラック番号 (NCN, HD)

DL=00H:MFM モード

=80H:FM モード

INT 80H

(作業領域)

BYTE[SURFMODE]

=サーフェスモード

BYTE[DISKMODE # 0/# 1]

=ドライブ # 0/# 1 のディスクモード

BYTE[RETRYNUM]

=リトライカウント

BYTE[BIOSMODE]: PARM

=ユーザパラメータ使用の有無

出力 AH=ステータスコード

AL =FDC の ST0

BH =FDC の ST1

BL =FDC の ST2

CH =FDC の C

CL =FDC の H

DH =FDC の R

DL =FDC の N

CF =エラーフラグ

(作業領域)

BYTE[RSLTSTAT]

=FDC のリード ID コマンド

BYTE[RSLTSTAT+1~+7]

=FDC の ST0~ST2, C, H, R, N

BYTE[SINTSTAT # 0/# 1]

=FDC の ST0(シーク時)

BYTE[SINTSTAT # 0/# 1+1]

=FDC の PCN(シーク時)

解説 指定されたトラックで最初に発見された正常な ID を読み出す。[BIOSMODE] の PARM ビットに 1 が立っている場合には、最大シリンダ番号として[USERPARM] の MAXCYL を用いる。またこのコマンドに限り、指定した FDC パラメータと読み出した ID が一致しなくてもエラーは返さない。

06H	リキャリブレイト
-----	----------

入力 AH=06H

CH=論理ドライブ番号 (US1, US0)

INT 80H

出力 AH=ステータスコード

CF=エラーフラグ

(作業領域)

BYTE[SINTSTAT # 0/# 1]

=FDC の ST0

BYTE[SINTSTAT # 0/# 1+1]

=FDC の PCN

解説 指定されたドライブのヘッドをリキャリブレイト(シリンダ 0 へ移動)する。FDC へのリキャリブレイトコマンドを 2 回発行したのち、一旦シリンダ 5 までシークし、再度リキャリブレイトコマンドを発行する。

07H	実行結果ステータスの取得
-----	--------------

入 力 AH=07H
CH=論理ドライブ番号 (US1, US0)
INT 80H

出 力 AH=ステータスコード
AL=FDC の ST3
CF=エラーフラグ

解 説 FDC に対して SENSE DEVICE STATUS コマンドを発行し、ST3 の値を受け取る。

08H	ドライブのレディチェック
-----	--------------

入 力 AH=08H
CH=論理ドライブ番号 (US1, US0)
INT 80H

出 力 AH=ステータスコード
CF=0: ドライブレディ
=1: ノットレディ

解 説 指定されたドライブがアクセス可能な状態にあるかどうかをチェックする。

09H	ドライブドアの開閉取得
-----	-------------

入 力 AH=09H
CH=論理ドライブ番号 (US1, US0)
INT 80H

出 力 AH=0
CF=0: 開閉無し
=1: 開閉の可能性あり

解 説 指定されたドライブに対して、ドアの開閉が行われたかどうかをチェックする。チェックする期間は、前回と同じドライブに対してこのコマンドが実行されてから、現時点までとする。モータ OFF の場合にはドアの開閉が判別できないため、CF=1 として返す。

0AH	ディスクモードの設定
-----	------------

入 力 AH=0AH
AL=ディスクモード
CH=論理ドライブ番号 (US1, US0)
INT 80H

出 力 AH=ステータスコード

CF=エラーフラグ

(作業領域)

BYTE[DISKMODE # 0/# 1]=ドライブ # 0/# 1 のディスクモード

解 説 ディスクモードを設定し、ドライブのハード的な諸条件を再設定する。
 <ディスクモード>

セクタ長 (バイト)		256	512	1024
1D/2D	系	01H	02H	03H
1DD/2DD	系	11H	12H	13H
1HD/2HD	系	21H	22H	23H
1HDs/2HDs	系	29H	—	—

<ハード的な設定>

- 1) ドライブの回転数の設定 (300rpm←→360rpm)
- 2) ドライブのトラック密度の設定 (48TPI←→96TPI)
- 3) FDC の動作クロックの設定 (4MHz←→8MHz)
- 4) FDC のスペシフィケーションの設定

ハード条件	1D/2D 系	1DD/2DD 系	1HD/2HD 系
回転数	300rpm	300rpm	360rpm
トラック密度	48TPI	96TPI	96TPI
動作クロック	4MHz	4MHz	8MHz
SRT, HUT	DAH	EAH	DFH
HLT, ND	18H	18H	30H

0BH	ワークエリアの初期化
-----	------------

入 力 AH=0BH

INT 80H

出 力 AH=0

CF=0

(作業領域)

BYTE[BIOSMODE]=0 (IDR=0, PARM=0, RAW=0)

BYTE[RETRYNUM]=8 (リトライ 8 回)

BYTE[MTOFFTIM]=250 (25 秒)

解 説 ユーザ用のワークエリアに正規の値を復帰させる。このコマンドは FDC に対するアクセスを一切行わない。

6.1.2 DISK BIOS 作業領域

当 DISK において使用される、主要な作業領域を以下に示す。作業領域は大別してコントロール領域とステータス領域に分かれる。コントロール領域は、DISK BIOS を呼び出す前にユーザ側で必要な設定を行うパラメータエリアであり、またステータス領域は DISK BIOS が内部で使用する領域であり、必要に応じてユーザ参照することができるが、ここをユーザ側で書き変えても BIOS の処理には影響を与えない。

作業領域はセグメント 0063H に置かれている。

●作業領域マップ

	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
+00H	a	b	予約域		c	d	e	f	未使用			g				
+10H	h								未使用			i	j	k	l	
+20H	m		n		予約域			未使用					o		p	
+30H	予 約															
+130H																

- ①コントロール領域
- a : [DISKMODE # 0] b : [DISKMODE # 1] c : [SURFMODE]
d : [BIOSMODE] e : [RETRYNUM] f : [MTOFFTIM]
g : [USERPARM]
- ②ステータス領域
- h : [RSLTSTAT] i : [UNITSTAT] j : [DINTSTAT]
k : [DOORSTAT] l : [MOTRSTAT] m : [SINTSTAT # 0]
n : [SINTSTAT # 1] o : [RETRYCNT] p : [MTOFFCNT]

(1) コントロール領域

+00H/ +01H	[DISKMODE # 0 / # 1]
---------------	----------------------

BYTE[+00H]=ドライブ # 0 のディスクモード
BYTE[+01H]=ドライブ # 1 のディスクモード

7	6	5	4	3	2	1	0
0	0	DSK1	DSK0	SPC	0	SCR1	SCR0

ビット	略 称	意 味	
B5	DSK1	B5	B4 媒体の種類
B4	DSK0	0	0 1D/2D 系
		0	1 1DD/2DD 系
		1	0 1HD/2HD 系
B3	SPC	B3	高密度媒体のフォーマット
B1	SCR1	0	1HD/2HD 系
		1	1HDs/2HDs 系
		B1	B0 セクタ長
B0	SCR0	0	1 256 バイト/セクタ
		1	0 512 バイト/セクタ
		1	1 1024 バイト/セクタ

内 容 ドライブ #0/#1 のディスクモードを格納する。ここは DOS と BIOS から共通に参照され、BIOS ではその上位 4 ビットのみが処理に影響する。初期化直後でディスクモードがまだ設定されていないときは 1D/2D のセクタ長 256 バイトモードの動作をする。

〈全モードの 16 進表示〉

セクタ長(バイト)		256	512	1024
1D/2D	系	01H	02H	03H
1DD/2DD	系	11H	12H	13H
1HD/2HD	系	21H	22H	23H
1HDs/2HDs	系	29H	—	—

ディスクは 1HD/2HDs を除いて、原則として MFM モードで書かれているものとする。またセクタ数とシリンドラ数は、ディスクモードから一意的に次のように決められる。

パラメータ		セクタ数				シリンドラ数
セクタ長		128	256	512	1024	—
1D/2D	系	—	16	09	05	40
1DD/2DD	系	—	16	09	05	80
1HD/2HD	系	—	26	15	08	80
1HDsm/2HDsm	系	—	26	—	—	80
1HDsf/2HDsf	系	26	—	—	—	—

上で添字 “f” が付くのは 1HDs/2HDs 系のトラック 0 を、“m” が付くのはそれ以外のトラックを表す。1HDs/2HDs 系のモードは、1HD/2HD：セクタ長 256 バイトのモードにおいて、さらに SPC ビットに 1 を立てることにより選択される。このモードが選択されると、トラック 0 のみを FM モード(セクタ長 128 バイト)として処理する。

+04H [SURFMODE]

7	6	5	4	3	2	1	0
0	0	0	0	0	0	SF#1	SF#0

SF#1 : ドライブ 1 サーフェスモード

SF#0 : ドライブ 0 サーフェスモード

0=片面モード

1=両面モード

+05H [BIOSMODE]

7	6	5	4	3	2	1	0
0	0	0	0	0	IDR	PARM	RAW

IDR : リード/ライトデータコマンドおよびフォーマットコマンドの入力条件の形式を指定する。

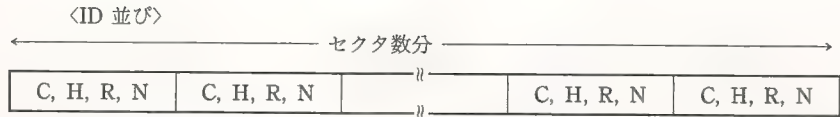
IDR=1 のときは次の意味になる。

1) リード/ライトデータ

IDR の C, H の値を、論理トラック番号と独立に指定する。

2) フォーマット

ユーザが用意した ID 並びに従って、フォーマットを行う。



PARM : PARM ビットに1が立つと, EOT, GPL 等の FDC パラメータに, ユーザが[USERPAR-M]に用意した5バイトの値を採用する。
RAW : RAW ビットに1が立つと, リードアフターライトを行う。

+06H	[RETRYNUM]
------	------------

リード／ライト系コマンドにおいてエラーが発生した場合の, リトライの回数を格納する。システムのデフォルト値としては8を取り, ファンクション 00H／0BH 実行時にこの値に再設定される。この領域を参照するファンクションの番号は 01H, 02H, 03H, 04H, 05H である。

+07H	[MTOFFTIM]
------	------------

コマンド終了からモータ OFF までの時間を格納する(100ms 単位)。システムのデフォルト値としては 250(25 秒)を取り, ファンクション 00H／0BH 実行時にこの値に再設定される。BIOS コマンドは実行時に無条件でモータ ON を行うため, 全てのコマンドがこの領域を参照する。

+0BH ~0FH	[USERPARM]
--------------	------------

- BYTE[+0BH]=最大シリンダ番号 (MAXCYL)
- BYTE[+0CH]=最大セクタ番号 (EOT)
- BYTE[+0DH]=リード／ライト用ギャップ長(RWGPL)
- BYTE[+0EH]=フォーマット用ギャップ長 (FMGPL)
- BYTE[+0FH]=リード／ライト用データ長 (DTL, 通常は FFH)

ユーザ用の FDC パラメータを格納する。これらのパラメータは, [BIOSMODE]の PARM ビットに1を立てることにより有効となる。
DISK BIOS での標準的 FDC パラメータは次のようになっている。

【FM モード】	1D／2D			1DD／2DD			1HD／2HD		
	128	256	512	128	256	512	128	256	512
MAXCYL	使用不可			使用不可			79	79	79
EOT							26	15	08
RWGPL							07H	0EH	1BH
FMGPL							1BH	2AH	3AH
DTL							80H	FFH	FFH

【MFM モード】	1D／2D			1DD／2DD			1HD／2HD		
	256	512	1024	256	512	1024	256	512	1024
MAXCYL	39	39	39	79	79	79	79	79	79
EOT	16	09	05	16	09	05	26	15	08
RWGPL	0EH	2AH	35H	0EH	2AH	35H	0EH	2AH	35H
FMGPL	36H	50H	74H	36H	50H	74H	36H	50H	74H
DTL	FFH	FFH	FFH	FFH	FFH	FFH	FFH	FFH	FFH

(2) ステータス領域

+10-17H [RSLTSTAT]

BYTE[+10H]=リード/ライト系 FDC コマンド (COM)
 BYTE[+11H]=FDC のリザルトステータス 0 (ST0)
 BYTE[+12H]=FDC のリザルトステータス 1 (ST1)
 BYTE[+13H]=FDC のリザルトステータス 2 (ST2)
 BYTE[+14H]=FDC の ID シリンダ番号 (C)
 BYTE[+15H]=FDC の ID ヘッド番号 (H)
 BYTE[+16H]=FDC の ID セクタ番号 (R)
 BYTE[+17H]=FDC の ID セクタ長 (N)

FDC の READ/WRITE 系コマンドにおけるリザルトフェーズの結果を格納する。この領域は BIOS コマンド内部で設定されるため、ここをユーザ側で変更してもコマンドの処理には影響しない。この領域を参照するファンクションの番号は 01H, 02H, 03H, 04H, 05H である。

+1CH [UNITSTAT]

7	6	5	4	3	2	1	0
0	0	0	0	0	0	UN #1	UN #0

UN #1 : ドライブ #1 の接続状況 0=開閉なし
 UN #0 : ドライブ #0 の接続状況 1=開閉あり

ドライブ #0/ #1 の接続の有無を表す。ファンクション 00H で設定される。ここをユーザが書き換えても BIOS の処理には影響しない。

+1DH [DINTSTAT]

7	6	5	4	3	2	1	0
0	0	0	0	0	0	IT #1	IT #0

IT #1 : ドライブ #1 の割込み状況 0=割込みなし
 IT #0 : ドライブ #0 の割込み状況 1=割込みあり

FDC の割込みルーチンにおいて、割込みのあったドライブのビットに 1 がセットされ、上位のルーチンでそのビットが検出後、0 にリセットされる。この領域は BIOS 全体から共通に設定/参照されるため、ここをユーザ側で変更してはならない。

+1EH [DOORSTAT]

7	6	5	4	3	2	1	0
0	0	0	0	0	0	DR #1	DR #0

DR #1 : ドライブ #1 の開閉状況 0=接続なし
 DR #0 : ドライブ #0 の開閉状況 1=接続あり

FDC の割込みルーチンにおいてドアの開閉が検出された場合、該当するドライブのビットに 1 がセットされる。このステータスにより、一定期間中のドアの開閉の有無を

知ることができる。ユーザがドライブのモータ OFF を行う場合には、この領域に 03H を設定する。

+1FH	[MOTRSTAT]
------	------------

FFH=モータ ON
00H=モータ OFF

モータの ON/OFF 状態を格納する。この領域は BIOS 内部からも参照されるため、ユーザ側でここを変更する場合には、実際のモータの設定も行う必要がある。またモータ ON/OFF の際には、必ず両ドライブを同一の状態に設定すること。さらにモータ OFF を行う場合には、[DOORSTAT]に 03H を出力する必要がある。

+20H ~23H	[SINTSTAT # 0/# 1]
--------------	--------------------

BYTE[+20H]=ドライブ # 0 のリザルトステータス (ST0)
BYTE[+21H]=ドライブ # 0 の物理シリンダ番号 (PCN)
BYTE[+22H]=ドライブ # 1 のリザルトステータス (ST0)
BYTE[+23H]=ドライブ # 1 の物理シリンダ番号 (PCN)

FDC の SEEK/RECALIBRATE コマンドにおけるリザルトフェーズの結果を格納する。ここをユーザ側で変更しても影響しない。

+2EH	[RETRYCNT]
------	------------

リード/ライト系コマンドにおけるリトライカウンタ。具体的には、コマンド実行時に [RETRYNUM]の値がこの領域にコピーされ、これがデクリメントされていく。したがって、ユーザ側でここを変更してもコマンドの処理には影響しない。

+2FH	[MTOFFCNT]
------	------------

モータ OFF 制御用のカウンタ。具体的には、各コマンドの終了時に [MTOFFTIM]の値がこの領域にコピーされ、タイマ割込み毎にこの値がデクリメントされていく。モータを強制的に停止させたい場合には、まずハード的にモータ OFF を行った後、この領域をゼロクリアし、ポート 01B6H に 00H を出力すること。最後の処理により、タイマ割込みが停止する。また単にハード的にモータを起動したい場合には、特にこの領域を設定する必要はない。

6.2 ハードディスク BIOS

ハードディスクを制御する BIOS であり、以下の機能を持っている。

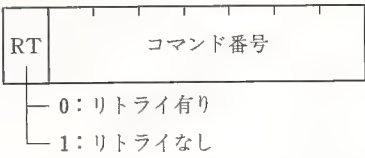
●ハードディスク BIOS 機能一覧

機能番号 (16 進)	機 能
00	ハードディスクシステムの初期化
01	データの読出し
02	データの書込み
03	1トラックフォーマット
04	1ドライブフォーマット
05	ヘッドのリキャリブレイト
06	ヘッドのリトラクト
07	代替トラックの用意
08	代替トラックの設定
09	ドライブ情報の取得
0A	作業領域アドレスの取得

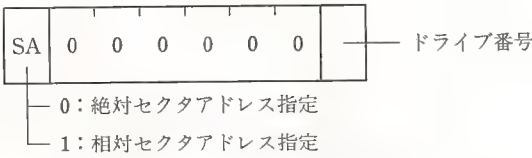
●呼出し時のパラメータ

各ファンクションを呼び出す時のパラメータの一般型は次の通りである。

AH=コマンド番号



BH=ドライブ番号(0または1)およびセクタアドレス指定方法



○絶対セクタアドレス指定のとき

CX=シリンダ番号

DH=ヘッド番号

DL=セクタ番号

	Cylinder #	Head #	Sec #
5MB	0-152	0-3	0-32
10MB	0-309	0-3	0-32
20MB	0-307	0-7	0-32
40MB	0-614	0-3	0-32

○相対セクタアドレス指定のとき

DL: CX 相対セクタ番号 = ((Cylinder * # HEAD + Head) * 33 + Sector)
HEAD = 4 (5MB or 10MB)
= 8 (20MB or 40MB)

ES: BP = リード/ライト時のデータバッファ

(注) 奇数番地でもよい。FFFFFH をまたぐ領域は設定できない。

AL = データ長/フォーマットデータ/ディスクモードなど

●呼出し方

- ① パラメータ設定
- ② AH ← 機能番号
- ③ INT 81H

レジスタは AX/CF 以外は値を返す場合を除いて保存される。

●ステータス

各ファンクションは終了時に AH レジスタにリザルトステータスを返す。エラー状態のときは同時にキャリーフラグをセットする。

CF=0 (ノーエラー)

AH の値(16進)

- 00H 正常終了
- 01H ライトプロテクト(SENSE 実行時のみ)

CF=1 (エラー)

AH の値(16進)

- 02H パラメータエラー
指定したパラメータがレンジ外のとき。
バッファエリアが FFFFFH をまたぐとき。
- 03H デバイス異常
ドライブの異常/ドライブが接続されていないなどの場合。
- 04H ノットレディ
ドライブがレディ状態にない場合。
- 05H ライトプロテクト
書き込み時にライトプロテクト状態が検出されたとき。
- 06H CRC エラー(ID)
ID の CRC エラーが検出されたとき。
- 07H CRC エラー(DATA)
DATA 部の CRC エラーが検出されたとき。
- 08H ノーセクタ
トラックに指定した ID が見つからないとき。
- 09H ノーシリング
エラー 08 の特殊な場合で、指定した ID のシリング番号が見つからないとき、シークエラーの可能性があり、RECALIBRATE の必要がある。

- 0AH 不良シリンダ
不良トラック指定があるトラックを読んだとき。
- 0BH ミッシングアドレスマーク (ID)
トラックに正常な ID がひとつも見つからないとき。
- 0CH ミッシングアドレスマーク (DATA)
ID のあとに正常な DATA 部が続かないとき。
- 10H 代替トラックが読めないとき
- 11H 代替トラックへ直接アクセスしている。
- 12H シークエラー
誤ったシリンダまたはヘッドを選択した場合
- FEH タイムアウトエラー
- FFH 不定エラー(その他のエラー)

6.2.1 ハードディスク BIOS ファンクション

00H	ハードディスクシステムの初期化
-----	-----------------

入 力 AH=00H
INT 81H

出 力 AH=00
CF=0
(作業領域)
BYTE[DSKTYPE#0/#1] =ドライブ#0/#1 ディスクタイプ
BYTE[DSKPRMADR#0/#1]=パラメータテーブルの先頭アドレス
BYTE[EQUIPOSK] =ドライブ接続状況
BYTE[HDINTFLG] =0
BYTE[COMPSTAT] =HDC コマンド
BYTE[COMPSTAT+1] =HDC 終了ステータス
BYTE[SCOMPSTAT] =エラー時の RESUEST SENSE の終了ステータス
BYTE[SENSESTAT+0~3] =エラー時の Result センスバイト

解 説 ハードディスク BIOS の初期化を行う。

1)HDC の初期化

HDC をリセットし、HDC がレディになったらディップスイッチの状態からドライブのタイプを判別し、HDC の ASSIGN DISK PARAMETER コマンドを用いてディスクの仕様を HDC にプログラミングする。

2)ドライブの接続状態のチェック

ディップスイッチからドライブの接続状態をチェックし、HDC の TEST DRIVE READY コマンドを各ドライブに対して発行し、レディ状態のドライブを[EQUIP-DSK]に設定する。

3)RECALIBRATE を実行後、RETRACT を行う。

01H	データの読出し
-----	---------

入 力 AH=01H/81H

AL =読み出すセクタ数

BH=セクタ指定形式(bit7)/ドライブ番号(bit0)

絶対セクタアドレス指定時(bit7=0):CX/DH/DL=Cyl/Hd/Sec

相対セクタアドレス指定時(bit7=1):DL:CX=セクタ番号

ES:BP=バッファ領域

INT 81H

(作業領域)

BYTE[DSKPRMADR # 0/# 1]=パラメータテーブルの先頭アドレス

出 力 AH=ステータス(エラー無のとき 00)

CF=エラーフラグ(エラー無のとき 0)

(作業領域)

BYTE[COMPSTAT] =HDC コマンド

BYTE[COMPSTAT+1] =HDC 終了ステータス

BYTE[SCOMPSTAT] =エラー時の RESUEST SENSE の終了ステータス

BYTE[SENSESTAT+0~3] =エラー時の Result センスバイト

解 説 HDC の READ コマンドにより、指定されたセクタを読み出し、ES:BP で指定されるバッファ領域に格納する。

02H	データの書込み
-----	---------

入 力 AH=02H/82H

AL =書き込むセクタ数

BH=セクタ指定形式(bit7)/ドライブ番号(bit0)

絶対セクタアドレス指定時(bit7=0):CX/DH/DL ← Cyl/Hd/Sec

相対セクタアドレス指定時(bit7=1):DL:CX ← セクタ番号

ES:BP=バッファ領域

INT 81H

(作業領域)

BYTE[DSKPRMADR # 0/# 1]=パラメータテーブルの先頭アドレス

出 力 AH=ステータス(エラー無のとき 00)

CF =エラーフラグ(エラー無のとき 0)

(作業領域)

BYTE[COMPSTAT] =HDC コマンド

BYTE[COMPSTAT+1] =HDC 終了ステータス

BYTE[SCOMPSTAT] =エラー時の RESUEST SENSE の終了ステータス

BYTE[SENSESTAT+0~3] =エラー時の Result センスバイト

解 説 HDC の WRITE コマンドにより、指定されるセクタに、ES:BP で示すバッファ領域のデータを書き込む。

03H	1トラックフォーマット
-----	-------------

入 力 AH=03H/83H

AL=インタリーブファクタ (01-10H)

BH=セクタ指定形式 (bit7)/ドライブ番号 (bit0)

絶対セクタアドレス指定時 (bit7=0): CX/DH=Cyl/Hd

相対セクタアドレス指定時 (bit7=1): DL: CX=セクタ番号

INT 81H

(作業領域)

BYTE[DSKPRMADR #0/#1]=パラメータテーブルの先頭アドレス

出 力 AH=ステータス (エラー無のとき 00)

CF=エラーフラグ (エラー無のとき 0)

(作業領域)

BYTE[COMPSTAT] =HDC コマンド

BYTE[COMPSTAT+1] =HDC 終了ステータス

BYTE[SCOMPSTAT] =エラー時の RESUEST SENSE の終了ステータス

BYTE[SENSESTAT+0~3] =エラー時の Result センスバイト

解 説 HDC の FORMAT TRACK コマンドにより, BH で指定されるドライブの CX, DX で指定されるトラックをフォーマットをする。

04H	1ドライブフォーマット
-----	-------------

入 力 AH=04H/84H

AL=インタリーブファクタ (01-10H)

BH=ドライブ番号 (bit0)

INT 81H

(作業領域)

BYTE[DSKPRMADR #0/#1]=パラメータテーブルの先頭アドレス

出 力 AH=ステータス (エラー無のとき 00)

CF=エラーフラグ (エラー無のとき 0)

(作業領域)

BYTE[COMPSTAT] =HDC コマンド

BYTE[COMPSTAT+1] =HDC 終了ステータス

BYTE[SCOMPSTAT] =エラー時の RESUEST SENSE の終了ステータス

BYTE[SENSESTAT+0~3] =エラー時の Result センスバイト

解 説 RECALIBRATE 後の FORMAT DRIVE コマンドにより, BH で指定されるドライブのフォーマットを行う。

05H	ヘッドのリキャリブレイト
-----	--------------

- 入 力** AH=05H/85H
BH=ドライブ番号(bit0)
INT 81H
(作業領域)
BYTE[DSKPRMADR # 0/# 1] =パラメータテーブルの先頭アドレス
- 出 力** AH=ステータス(エラー無のとき 00)
CF=エラーフラグ(エラー無のとき 0)
(作業領域)
BYTE[COMPSTAT] =HDC コマンド
BYTE[COMPSTAT+1] =HDC 終了ステータス
BYTE[SCOMPSTAT] =エラー時の RESUEST SENSE の終了ステータス
BYTE[SENSESTAT+0~3] =エラー時の Result センスバイト
- 解 説** BH で指定されるドライブのヘッドをシリンダ 0 へ移動させる。SEEK コマンドにより一旦シリンダ 5 へシークさせた後、RECALIBRATE コマンドによりトラック 0 に移動する。

06H	ヘッドのリトラクト
-----	-----------

- 入 力** AH=06H/86H
BH=ドライブ番号(bit0)
INT 81H
(作業領域)
BYTE[DSKPRMADR # 0/# 1] =パラメータテーブルの先頭アドレス
- 出 力** AH=ステータス(エラー無のとき 00)
CF=エラーフラグ(エラー無のとき 0)
(作業領域)
BYTE[COMPSTAT] =HDC コマンド
BYTE[COMPSTAT+1] =HDC 終了ステータス
BYTE[SCOMPSTAT] =エラー時の RESUEST SENSE の終了ステータス
BYTE[SENSESTAT+0~3] =エラー時の Result センスバイト
- 解 説** BH で指定されるドライブのヘッドを不必要シリンダへ移動させる。
1) ASSIGN DISK PARAMETER コマンドで RETRACT 用のパラメータを設定する。
2) SEEK コマンドで MCA(maximum cylinder address)まで移動する。
※通常アクセス時とリトラクト時の MCA は異なる。
3) ASSIGN DISK PARAMETER コマンドで通常アクセス用のパラメータに再設定する。

07H	代替トラックの用意
-----	-----------

入 力 AH=07H

BH=セクタ指定形式(bit7)/ドライブ番号(bit0)

絶対セクタアドレス指定時(bit7=0):CX/DH=Cyl/Hd

相対セクタアドレス指定時(bit7=1):DL:CX=セクタ番号

ES:BP=4 バイトのバッファ領域

INT 81H

(作業領域)

BYTE[DSKPRMADR #0/#1]=パラメータテーブルの先頭アドレス

出 力 AH=ステータス(エラー無のとき 00)

CF=エラーフラグ(エラー無のとき 0)

ES:[BP+0~3]=代替トラックアドレス

解 説 指定されたトラックの相対セクタアドレスを ES:BP で指定されたバッファ領域に格納する。ここで指定されたトラックは次のファンクション 08H で指定された不良トラックの代替トラックとして使用される。

08H	代替トラックの設定
-----	-----------

入 力 AH=08H

BH=セクタ指定形式(bit7)/ドライブ番号(bit0)

絶対セクタアドレス指定時(bit7=0):CX/DH=Cyl/Hd

相対セクタアドレス指定時(bit7=1):DL:CX=セクタ番号

ES:BP=コマンド 7 で設定した 4 バイトのバッファ領域

INT 81H

(作業領域)

BYTE[DSKPRMADR #0/#1]=パラメータテーブルの先頭アドレス

出 力 AH=ステータス(エラー無のとき 00)

CF=エラーフラグ(エラー無のとき 0)

BYTE[COMPSTAT] =HDC コマンド

BYTE[COMPSTAT+1] =HDC 終了ステータス

BYTE[SCOMPSTAT] =エラー時の RESUEST SENSE の終了ステータス

BYTE[SENSESTAT+0~3] =エラー時の Result センスバイト

解 説 HDC の ASSIGN ALTERNATE TRACK コマンドにより、BH で指定されるドライブの CX, DX で指定される不良トラックの代替トラックとして、ファンクション 07H で指定された代替トラックを割り当てる。

09H	ドライブ情報の取得
-----	-----------

入 力 AH=09H

BH = ドライブ番号 (bit0)

INT 81H

(作業領域)

BYTE[DSKPRMADR # 0/# 1] = パラメータテーブルの先頭アドレス

出 力 AH = ステータス (エラー無のとき 00)

CF = エラーフラグ (エラー無のとき 0)

AL = ディスクタイプ

0 = 5MB

1 = 10MB

2 = 20MB

3 = 40MB

(作業領域)

BYTE[COMPSTAT] = HDC コマンド

BYTE[COMPSTAT+1] = HDC 終了ステータス

BYTE[SCOMPSTAT] = エラー時の RESUEST SENSE の終了ステータス

BYTE[SENSESTAT+0~3] = エラー時の Result センスバイト

解 説 HDC の TEST DRIVE READY コマンドにより, BH で指定されるドライブの状態を通知する。

0AH	作業領域アドレスの取得
-----	-------------

入 力 AH=0AH

INT 81H

出 力 AH=00

CF = 0

ES:BP = ハードディスク BIOS 用大域変数領域アドレス

解 説 BIOS で使用している作業領域 (大域変数領域) の先頭アドレスを ES:BP に返す。

6.2.1 ハードディスク BIOS 作業領域

当 BIOS で使用される主な作業領域を以下に示す。セグメントアドレスは 0077H である。

	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
+00	a	b	c				d	e	f	g	h		i		-	
a :	[COMPSTAT]								f : [DSKTYPE#0]							
b :	[SCOMPSTAT]								g : [DSKTYPE#1]							
c :	[SENSESTAT]								h : [DSKPRMADR#0]							
d :	[HDINTFLG]								i : [DSKPRMADR#1]							
e :	[EQUIPSK]															

+00H COMPSTAT (2Bytes)

HDC の各コマンドの終了時に HDC から送られてくる終了ステータスを格納する。

[+00H] : HDC へ送ったコマンド

[+01H] : HDC コマンドに対する終了ステータス

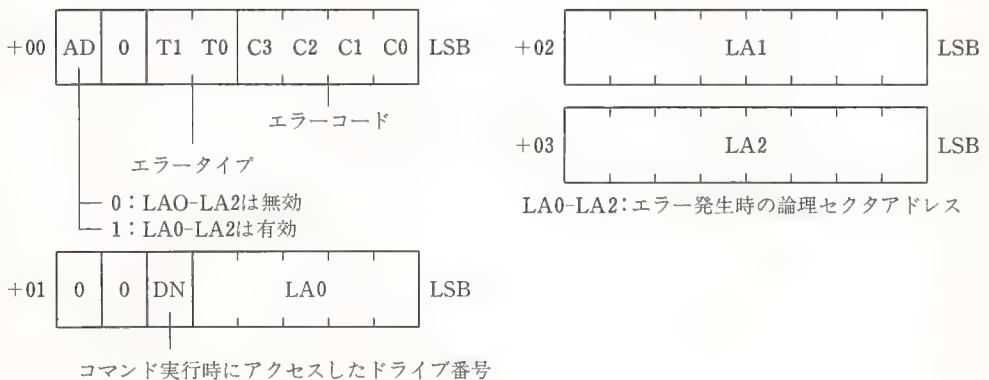


+02H SCOMPSTAT (1Byte)

エラー発生時に発行する REQUEST SENSE コマンドの終了ステータスを格納する。

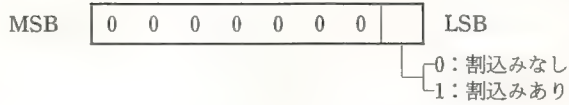
+03H SENSESTAT (4Bytes)

エラー発生時に発行する REQUEST SENSE コマンドの RESULT SENSE を格納する。



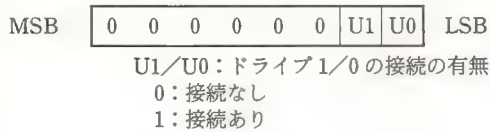
+07H	HDINTFLG(1Byte)
------	------------------------

HDCの割込みが発生すると、割込みルーチンで1に設定される。リセットは上位ルーチンが行う。



+08H	EQUIPDSK(1Byte)
------	------------------------

ドライブ #0/#1 の接続の有無を表す。ファンクション 00H で設定される。

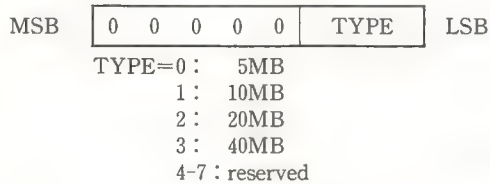


+09H/ 0AH	DSKTYPE #0/#1(1Byte each)
--------------	----------------------------------

ドライブ #0/#1 のディスクタイプを表す。ファンクション 00H で設定される。

[+09H]: ドライブ 0 のディスクタイプ

[+0AH]: ドライブ 1 のディスクタイプ



+0BH/ +0DH	DSKPRMADR #0/#1(2Bytes each)
---------------	-------------------------------------

ハードディスク BIOS 内のディスクパラメータテーブルの先頭アドレス(オフセット)を格納する。

[+0BH]: ドライブ 0 の Disk parameter table へのポインタ

[+0DH]: ドライブ 1 の Disk parameter table へのポインタ

6.3 キーボード BIOS

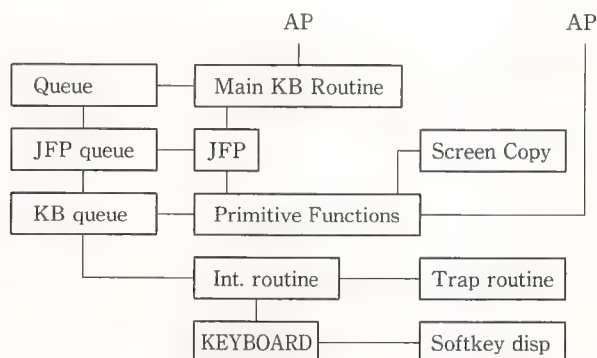
キーボード BIOS(以降 KB BIOS)はキーボードからの入力をバッファリングした後、そのまま/または JFP を通してアプリケーションプログラムへ渡す機能をもつ。また画面ハードコピーの起動、ポップアップの起動の制御も行う。

●キーボード BIOS 機能一覧

機能番号 (16 進)	機 能
00	1 文字入力
01	キューのセンス
02	ソフトキー展開の指定
03	ソフトキー文字列の設定
04	作業領域の初期化
05	KB BIOS の初期化
06	ポップアップトラップの登録
07	ポップアップトラップの削除
08	ユーザトラップの登録・削除
09	キーコードの 1 文字入力
0A	KB キューのセンス
0B	シフト類キー押下状態のセンス
0C	KB キューのクリア
0D	キーコードグループ押下状態のセンス
0E	基本ファンクション作業領域アドレスの取得
0F	キーリピート機能設定
10	画面コピー機能設定

6.3.1 キーデータフロー

キーボードから入力されたデータは下図の下から上へと変換を受けながら渡されていく。



○割込みルーチン(Int. routine)

キーボードから送られてきたデータを KB キューに入れ、必要なワークを設定する。
キューがフルになると BEEP 音を発生する。

○基本ファンクション(Primitive Functions)

KB queue のデータをアプリケーションソフトまたは JFP の要求に従い取り出す。

○日本語入力フロントエンドプロセッサ(JFP)

キー操作または標準ファンクションの要求に従い基本ファンクションからのデータを日本語変換して(または変換せずに)、標準ファンクションルーチンへ渡す。

○標準ファンクション(Main KB Routine)

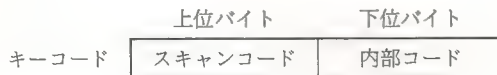
アプリケーションソフトの要求に従い、JFP からのデータを取り出す。ソフトキーの展開もこなう。

○トラップルーチン

ポップアップアプリケーション機能を実現するために、キーが押されたときに呼び出されるルーチンを登録することができる。

6.3.2 キーコード

キーコードとはキーの押下に対応して KB キューに入れられ、基本ファンクションで取り出される 2 バイトのコードであり、スキャンコードと内部コードとからなる。



(1) スキャンコード

スキャンコードはキーボードの各キーに一意に割り当てられたコードである。ソフトキー以外のキーのスキャンコードはシフト状態によって変化しない。スキャンコードの最大値は B5H である。

(2) 内部コード

内部コードはいわゆる ASCII コードであり、88M/F シリーズのキーコードと共通である。内部コードの最大値は F7H である。

(3) ユーザ定義コード

トラップルーチンから実際に存在しない仮想的なキーのキーコードを返すために FF01H から FFFE H の 254 種類のユーザ定義キーコードを使用することができる。このユーザ定義キーコードに対応する 2 バイトコードは必ず 00FFH となる。

6.3.3 その他

(1) 画面のハードコピー

COPY キーが押されると割り込みルーチンは画面コピーフラグを設定するが、実際には画面コピーは行わない。基本ファンクションの #09/#0A が呼び出されたときに画面コピーフラグが設定されていると実際に画面コピーを行う。逆に KB BIOS を呼ばないアプリケーションソフトでは自動的な画面コピーは行われぬ。この場合、アプリケーションソフトが KB BIOS の画面コピーフラグを積極的に読み出して画面コピーファンクションを呼ぶことにより行うことができる。

なお COPY キーのキーコードがキューに入ることはない。

(2) ソフトキー

ソフトキーとは一つのキーに対して自由に文字列を割り当てることができるキーであり、VA では以下のキーを指す。

f1-f10, f11-f20(shift+fx)

INS, DEL, 矢印キー, HOME, CLR, HELP, ROLL UP, ROLL DOWN

ソフトキーの設定／押下時の文字列取出しは標準ファンクションでおこなう。ソフトキーの表示、シフトキーが押されたときの表示の変更は割込みルーチンの指示によりテキスト BIOS が行う。また、JFP もソフトキーの表示に変更を加える。

(3) シフト状態の取得

アプリケーションソフトがシフト類キーの状態を知るためには次の方法がある。

- ① IN 命令によりハードウェアからキーの状態を読み出す
- ② KB BIOS ファンクションを使って各シフト類キーの状態を知る
- ③ KB BIOS の作業領域をみてキーの状態を読み出す
- ④ シフト類キーの状態によって変化するいくつかのキーコードを用いる

6.3.4 標準ファンクション

標準ファンクションは JFP を通してデータを受け取る。ただし JFP がオフのときは日本語変換は行われない。

- 00H 2 バイトコードで1文字取り出す(入力待ちあり)
- 01H キューをセンスする
- 02H ソフトキーの展開の有無を指定する。表示とは無関係
- 03H ソフトキー文字列を設定する
- 04H キューその他のワークを初期化する
- 05H KB BIOS を初期状態に戻す

00H	1 文字入力
-----	--------

入 力 AH=00H
INT 82H

出 力 AX=展開ありのとき：2 バイトコード
展開なしのとき：2 バイトコード or 0(ソフトキー)
BH=スキャンコード
BL=内部コード

解 説 キューから1文字取り出す。キューが空のときは JFP が文字を返すまで待つ。日本語変換が行われて対応するキーがない文字が返される場合などには、BX が FFFFH となる。

AX のレンジは、
 0000-00FCH ANK 文字
 8140-FCFCH 日本語文字
 00FFH ユーザー定義キーコード
 である。

ソフトキーが押されたときのこのファンクションの動作は展開指定によって異なる。展開無しの場合には、AX は 00H となり、BX にはキーコードが入る。展開を行うときには、AX には定義された文字が入り、BX は FFFFH となる。

ユーザ定義キー(トラップルーチンから返される)場合には、AX は 00FFH となり、BX は FF01-FFFEH となる。

01H	キューのセンス
-----	---------

入 力 AH=01H
INT 82H

出 力 AX=2 バイトコード or 0
BH=スキャンコード
BL=内部コード
CY=1 文字なし
=0 文字あり

解 説 キューの先頭文字をみる。キューの状態は変化しない。また、キューが空の場合でも入力待ちは行わない。

02H	ソフトキー展開の指定
-----	------------

入 力 AH=02H
AL=0(展開指定)
=1(非展開指定)
INT 82H

出 力 なし

解 説 ソフトキーの展開を行うかどうかを指定する。この指定はファンクション 00H/01H に関係する。また、ファンクションキーの表示とは全く無関係である。

03H	ソフトキー文字列の設定
-----	-------------

入 力 AH=03H
CL=ソフトキー番号
ES:DX=定義するストリング(ASCIZ 文字列)
INT 82H

出 力 CY=1 ソフトキー番号が異常
=0 正常終了

ソフトキー番号は以下のとおり

0:	全てのソフトキー
1-0AH:	f1~f10
0B-14H:	shift+f1~f10
15-16H:	Rollup/Roiidown
17-18H:	Ins/Del
19-1CH:	Up/Left/Right/Down keys
1DH:	Clr key
1EH:	Help key
1FH:	Home key

解 説 ソフトキーに文字列を定義する。文字列の長さは、ターミネータを含めて、ファンクションキーでは 20 バイト、その他のキーは 6 バイトである。ターミネータとして NULL (00) を文字列の最後に付加する必要がある。
文字コードとしては混在文字コードを用いる。

04H	作業領域の初期化
-----	----------

入 力 AH=04H
INT 82H

出 力 なし

解 説 キューをクリアするだけでなく、キーボード関係(JFP を含む)の設定をすべて初期状態に戻す。

05H	KB BIOS の初期化
-----	--------------

入 力 AH=05H
INT 82H

出 力 なし

解 説 キーボード BIOS 全体を初期化し、ハードウェアリセット直後と同様の状態にする。以下の初期化処理が行われる。

- 1) キューのクリア
- 2) ソフトキー文字列の初期化

f1:	<Load">	f11-f20:	xxxxxxx
f2:	<Auto>	ins:	12H
f3:	<Goto>	del:	7FH
f4:	<List>	left arrow:	1CH
f5:	<Run>+CR	right arrow:	1DH
f6:	<Save">	up arrow:	1EH

f7: <Key>	down arrow: 1FH
f8: <Print>	home: 0BH
f9: <Edit.>+CR	clr: 0CH
f10: <Cont>+CR	help: 01H
	roll up: 0F8H
	roll down: 0F9H

- 3) COPY キーエントリの初期化
- 4) トラップアドレスの初期化
- 5) ソフトキー展開あり
- 6) JFP の初期化

6.3.5 トラップ制御ファンクション

KB BIOS ではシフト類以外のキーが押されたときにあらかじめ設定されたルーチンを起動することができる。これをトラップ機能とよぶ。トラップは KB 割込みルーチンで発生する。8つのポップアップトラップと1つのユーザトラップが登録できる。

- 06H ポップアップトラップを登録する
- 07H ポップアップトラップを削除する
- 08H ユーザトラップを登録／削除する

合計9つのトラップには優先順位が設けられており、優先順位が高いトラップから順に呼び出される。ポップアップトラップ内での優先順位は、最後に定義されたルーチンが最高優先順位を持っている。ユーザトラップは最低優先順位である。また、トラップルーチンは CLI の状態で呼ばれる。

- ・トラップルーチンが呼ばれる時には、AX, CL には次のような値が入っている。

AH = スキャンコード
AL = 内部コード
CL = シフト状態

- ・トラップルーチンを終了するときには次のレジスタを設定する。

AH = スキャンコード
AL = 内部コード

このとき AX にユーザ定義キーコードを設定することもできる。
ただし、コードとしては必ず FF01-FFFEH を使用すること。

DL = KB BIOS への指示

このレジスタによって、リターン後の KB BIOS への指示を出すことができる。この処理は次優先順位のトラップが呼び出される前に行われる。

bit 7-3: Reserved (0 に設定)

bit 2: 低優先のトラップスキップ (0: スキップしない / 1: する)

bit 1: キューへ入れる (0: 入れない / 1: 入れる)

bit 0: キュークリア (0: しない / 1: する)

06H	ポップアップトラップの登録
-----	---------------

入 力 AH=06H
 ES:DX=トラップルーチンのアドレス
 INT 82H

出 力 AL=0-7:ポップアップトラップ番号
 =FFH:トラップベクタに空きがない

解 説 ポップアップトラップルーチンを登録する。登録できた場合には、ポップアップトラップ番号を返す。これは DOS のファイルハンドルと同様の論理的管理番号である。

07H	ポップアップトラップの削除
-----	---------------

入 力 AH=07H
 AL=ポップアップトラップ番号(0-7)
 INT 82H

出 力 AL=00H:正常終了
 =FFH:指定されたトラップが登録されていない

解 説 登録されているポップアップトラップルーチンを削除する。

08H	ユーザトラップの登録・削除
-----	---------------

入 力 AH=08H
 AL=0:ユーザトラップ削除
 =1:ユーザトラップ登録
 ES:DX=トラップルーチンのアドレス(AL=1のとき)
 INT 82H

出 力 なし

解 説 ユーザトラップルーチンの登録/削除を行う。

6.3.6 基本ファンクション

基本的なキーボード入力機能をサポートしたファンクション群であり、日本語変換は行わない。

09H KB キューからキーコードで1文字取り出す(入力待ち)

0AH KB キューをセンスする

0BH シフト類キーの押下状態をセンスする

0CH KB キューをクリアする

0DH キーコードグループの押下状態をセンスする

0EH 基本ファンクションの作業領域アドレスを得る

0FH キーリピート機能の ON/OFF

10H 画面コピー機能の ON/OFF

09H	キーコードの1文字入力
-----	-------------

入 力 AH=09H

INT 82H

出 力 AH=スキャンコード

AL=内部コード

解 説 KB キューから1文字取り出す。キューが空のときはキーボード入力を待つ。

0AH	KB キューのセンス
-----	------------

入 力 AH=0AH

INT 82H

出 力 AH=スキャンコード

AL=内部コード

CY=0: キー入力なし

=1: キー入力あり

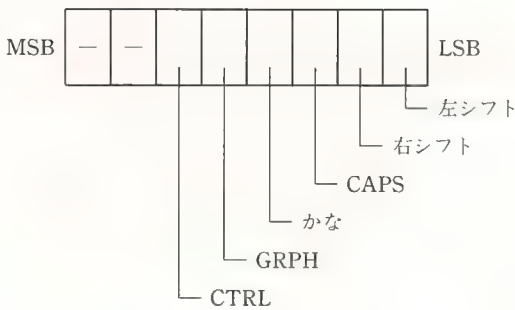
解 説 KB キューの先頭をみる。キューの状態は変化しない。キューが空のときは CY=0 で戻り、入力待ちはしない。

0BH	シフト類キー押下状態のセンス
-----	----------------

入 力 AH=0BH

INT 82H

出 力 AL=シフト状態(各ビット=1: 押下状態)



解 説 シフト類キーの押下状態をセンスする。

0CH	KB キューのクリア
-----	------------

入 力 AH=0CH
INT 82H

出 力 なし

解 説 キーボードとキューを初期化する。

0DH	キーコードグループ押下状態のセンス
-----	-------------------

入 力 AH=0DH
AL=キーコードグループ番号(I/O ポート 00～0EH のポート番号と同じ)
INT 82H

出 力 AH=押下キービットを1にしたデータ

解 説 指定されたキーコードグループの押下情報をビットマップで返す。

0EH	作業領域アドレスの取得
-----	-------------

入 力 AH=0EH
INT 82H

出 力 ES:DX=当 BIOS の作業領域の開始アドレス

0FH	キーリピート機能設定
-----	------------

入 力 AH=0FH
AL=0:オートリピート ON
≠0:オートリピート OFF
INT 82H

出 力 なし

解 説 キーボードのオートリピートの有無を指定する。初期値は ON である。

10H	画面コピー機能設定
-----	-----------

AH-10H

AL=FFH：画面コピー OFF

=0-7：画面コピーモード設定

INT 82H

画面コピーモード

MSB	0	0	0	0	0	SD	RL	BW	LSB
-----	---	---	---	---	---	----	----	----	-----

SD =0：カラーモード

=1：モノクロモード

RL =0：左右正常

=1：左右反転

BW =0：ノーマルモノクロモード

=1：リバースモノクロモード

出 力 なし

解 説 画面コピー機能の有無とタイプを指定する。

6.3.7 キーボード BIOS 作業領域

当 BIOS で使用される主なワークエリアを以下に示す。

セグメントアドレス ABH オフセット長さ(バイト)		内 容
0000	01	シフトキーの状態
0001	16×20	ファンクションキーに定義された文字列
0141	06×11	他のソフトキーの定義された文字列
0184	04	ユーザトラップのアドレス
0188	01	ユーザトラップフラグ
0189	01	展開フラグ(0=ON/1=OFF)
018A	02	ソフトキー文字列へのポインタ
018C	02	ソフトキーコード退避用
018E	01	コピーキーフラグ
018F	01	画面コピー制御モード
0190	01	予約
0191	32	キーコードバッファ(キュー)
01B1	01	定義されているポップアップトラップの数
01B2	04×8	ポップアップトラップのアドレス
01D2	01×8	ポップアップトラップの優先順位
01DA	04	KB BIOS ディスパッチアドレス(JFP 用)
01DE	04	JFP ディスパッチアドレス(KB BIOS 用)
01E2	04	KB BIOS HOOK # 1
01E6	04	KB BIOS HOOK # 2
01EA	04	KB BIOS HOOK # 3
01EE	04	KB BIOS HOOK # 4
01F2	04	KB BIOS HOOK # 5
01F6	01	画面コピー中止フラグ
01F7	01	ポート 152H のコピー

6.3.8 スキャンコード・内部コード対応表

スキャン コード	Key	Shift State	Bas	Shift (R)	Shift (L)	Caps	Caps Shift	カナ	カナ Shift	Grph	Ctrl
00	E S C		1B	1B	1B	1B	1B	1B	1B	1B	
01	! 1 ヌ		31	21	21	31	21	C7	C7		
02	" 2 フ		32	22	22	32	22	CC	CC		
03	# 3 ア ア		33	23	23	33	23	B1	A7		
04	\$ 4 ウ ウ		34	24	24	34	24	B3	A9		
05	% 5 エ エ		35	25	25	35	25	B4	AA	F2	
06	& 6 オ オ		36	26	26	36	26	B5	AB	F3	
07	- 7 ヤ ヤ		37	27	27	37	27	D4	AC	F4	
08	(8 ュ ユ		38	28	28	38	28	D5	AD	F5	
09) 9 ヨ ヨ		39	29	29	39	29	D6	AE	F6	
0A	0 フ ワ		30	30	30	30	30	DC	A6	F7	
0B	= 2D — 3D ホ		2D	3D	3D	2D	3D	CE	CE	8C	
0C	^ 5E ~ 5E		5E	5E	5E	5E	5E	CD	CD	8B	1E
0D	5C ¥ 7C — 7C		5C	7C	7C	5C	7C	B0	B0	F1	1C
0E	B S		08	08	08	08	08	08	08	08	08
0F	T A B		09	09	09	09	09	09	09	09	09

スキャンコードと内部コード対応表

スキャン コード	Key	Shift State	Bas	Shift (R)	Shift (L)	Caps	Caps Shift	カナ	カナ Shift	Grph	Ctrl
10	Q	タ	71	51	51	51	71	C0	C0	9C	11
11	W	テ	77	57	57	57	77	C3	C3	9D	17
12	E	イ イ	65	45	45	45	65	B2	A8	E4	05
13	R	ス	72	52	52	52	72	BD	BD	E5	12
14	T	カ	74	54	54	54	74	B6	B6	EE	14
15	Y	ン	79	59	59	59	79	DD	DD	EF	19
16	U	ナ	75	55	55	55	75	C5	C5	F0	15
17	I	ニ	69	49	49	49	69	C6	C6	E8	09
18	O	ラ	6F	4F	4F	4F	6F	D7	D7	E9	0F
19	P	セ	70	50	50	50	70	BE	BE	8D	10
1A	～ @	^	40	7E	7E	40	7E	DE	DE	8A	00
1B	[.	{	5B	7B	7B	5B	7B	DF	A2		1B
1C	Retrun		0D	0D	0D	0D	0D	0D	0D	0D	0D
1D	A	チ	61	41	41	41	61	C1	C1	9E	01
1E	S	ト	73	53	53	53	73	C4	C4	9F	13
1F	D	シ	64	44	44	44	64	BC	BC	E6	04

スキャンコードと内部コード対応表

スキャン コード	Key	Shift State	Bas	Shift (R)	Shift (L)	Caps	Caps Shift	カナ	カナ Shift	Grph	Ctrl
20	F	ハ	66	46	46	46	66	CA	CA	E7	06
21	G	キ	67	47	47	47	67	B7	B7	EC	07
22	H	ク	68	48	48	48	68	B8	B8	ED	08
23	J	マ	6A	4A	4A	4A	6A	CF	CF	EA	0A
24	K	ノ	6B	4B	4B	4B	6B	C9	C9	EB	0B
25	L	リ	6C	4C	4C	4C	6C	D8	D8	8E	0C
26	+		3B	2B	2B	3B	2B	DA	DA	89	
	;	レ									
27	*		3A	2A	2A	3A	2A	B9	B9	94	
	:	ケ									
28	」]	5D	7D	7D	5D	7D	D1	A3		1D
	「	ム									
29	Z	ツ	7A	5A	5A	5A	7A	C2	AF	80	1A
		ツ									
2A	X	サ	78	58	58	58	78	BB	BB	81	18
2B	C	ソ	63	43	43	43	63	BF	BF	82	03
2C	V	ヒ	76	56	56	56	76	CB	CB	83	16
2D	B	コ	62	42	42	42	62	BA	BA	84	02
2E	M	ミ	6E	4E	4E	4E	6E	D0	D0	85	0E
2F	M	モ	6D	4D	4D	4D	6D	D3	D3	86	0D

スキャンコードと内部コード対応表

スキャン コード	Key	Shift State	Bas	Shift (R)	Shift (L)	Caps	Caps Shift	カナ	カナ Shift	Grph	Ctrl
30	< ,	、 ネ	2C	3C	3C	2C	3C	C8	A4	87	
31	> .	。 ル	2E	3E	3E	2E	3E	D9	A1	88	
32	? /	。メ	2F	3F	3F	2F	3F	D2	A5	97	
33	-	ロ		5F	5F		5F	DB	DB		1F
34	(S P E C E)		20	20	20	20	20	20	20	20	20
35	X F E R (変換)		35 20	A5 20	A5 20	35 20	A5 20	35 20	A5 20	35 20	B5 20
36	R O L L U P		00	00	00	00	00	00	00	00	00
37	R O L L D O W N		00	00	00	00	00	00	00	00	00
38	I N S		00	00	00	00	00	00	00	00	00
39	D E L		00	00	00	00	00	00	00	00	00
3A	↑		3A 00	AA 00	AA 00	3A 00	AA 00	AA 00	AA 00	3A 00	BA 00
3B	←		3B 00	AB 00	AB 00	3B 00	AB 00	3B 00	AB 00	3B 00	BB 00
3C	→		3C 00	AC 00	AC 00	3C 00	AC 00	3C 00	AC 00	3C 00	BC 00
3D	↓		3D 00	AD 00	AD 00	3D 00	AD 00	3D 00	AD 00	3D 00	BD 00
3E	H O M E C L R		3E 00	AE 00	AE 00	3E 00	AE 00	3E 00	AE 00		
3F	H E L P		00	00	00	00	00	00	00	00	00

スキャンコードと内部コード対応表

スキャン コード	Key	Shift State	Bas	Shift (R)	Shift (L)	Caps	Caps Shift	カナ	カナ Shift	Grph	Ctrl
40	—		2D	2D	2D	2D	2D	2D	2D	2D	2D
41	/		2F	2F	2F	2F	2F	2F	2F	2F	2F
42	7		37	37	37	37	37	37	37	98	37
43	8		38	38	38	38	38	38	38	91	38
44	9		39	39	39	39	39	39	39	99	39
45	*		2A	2A	2A	2A	2A	2A	2A	95	2A
46	4		34	34	34	34	34	34	34	E1	34
47	5		35	35	35	35	35	35	35	E2	35
48	6		36	36	36	36	36	36	36	E3	36
49	+		2B	2B	2B	2B	2B	2B	2B	E0	2B
4A	1		31	31	31	31	31	31	31	93	31
4B	2		32	32	32	32	32	32	32	8 F	32
4C	3		33	33	33	33	33	33	33	92	33
4D	=		3D	3D	3D	3D	3D	3D	3D	96	3D
4E	0		30	30	30	30	30	30	30	9 A	30
4F	,		2C	2C	2C	2C	2C	2C	2C	90	2C

スキャンコードと内部コード対応表

スキャン コード	Key	Shift State	Bas	Shift (R)	Shift (L)	Caps	Caps Shift	カナ	カナ Shift	Grph	Ctrl
50	.		2E	2E	2E	2E	2E	2E	2E	9B	2E
51	N F E R (決定)		51 20	A1 20	A1 20	51 20	A1 20	51 20	A1 20	51 20	B1 20
60	STOP		03	03	03	03	03	03	03	03	03
61	COPY										
62	F・1		62 00	82 00	82 00	62 00	82 00	62 00	82 00		92 00
63	F・2		63 00	83 00	83 00	63 00	83 00	63 00	83 00		93 00
64	F・3		64 00	84 00	84 00	64 00	84 00	64 00	84 00		94 00
65	F・4		65 00	85 00	85 00	65 00	85 00	65 00	85 00		95 00
66	F・5		66 00	86 00	86 00	66 00	86 00	66 00	86 00		96 00
67	F・6		67 00	87 00	87 00	67 00	87 00	67 00	87 00		97 00
68	F・7		68 00	88 00	88 00	68 00	88 00	68 00	88 00		98 00
69	F・8		69 00	89 00	89 00	69 00	89 00	69 00	89 00		99 00
6A	F・9		6A 00	8A 00	8A 00	6A 00	8A 00	6A 00	8A 00		9A 00
6B	F・10		6B 00	8B 00	8B 00	6B 00	8B 00	6B 00	8B 00		9B 00

スキャンコードと内部コード対応表

スキャン コード	KeyShift State	Bas	Shift (R)	Shift (L)	Caps	Caps Shift	カナ	カナ Shift	Grph	Ctrl
70	SHIFT(L)									
71	CAPS									
72	GRAPH									
73	CTRL									
78	SHIFT(R)									
79	RETURN TOKEY	0D	0D	0D	0D	0D	0D	0D	0D	0D
7A	PC	00	00	00	00	00	00	00	00	00
7B	全角	00	00	00	00	00	00	00	00	00

6.4 テキスト BIOS

テキスト BIOS は VA のテキスト表示機能を制御する BIOS であり、以下の 44 個の機能がある。

●テキスト BIOS 機能一覧

機能番号 (16 進)	機 能
00	1 文字表示 (コントロールコード有)
01	1 文字表示 (コントロールコード無)
02	ASCIZ 文字列の表示
05	カレントアトリビュートの設定
06	画面へのアトリビュート設定
07	カラー設定
08	カーソル位置の設定
09	カーソル上移動
0A	カーソル下移動
0B	カーソル右移動
0C	カーソル左移動
0D	カーソル下移動 (スクロール付)
0E	カーソル上移動 (スクロール付)
10	水平スクロール
11	TAB 位置の設定
12	TAB 位置の解除
13	行削除
14	文字削除
15	行挿入
16	文字挿入
17	表示画面のクリア
18	行のクリア
19	フレームバッファのクリア
1A	フレームバッファ中の位置取得
1B	カレントスクリーンの変更
1C	フレームバッファの定義
1D	分割画面の設定
1E	POPUP スクリーンのオープン
1F	POPUP スクリーンのクローズ
20	POPUP スクリーンの画面モード変更
21	外字フォントの設定
22	外字フォントの削除
23	文字フォントの取得
24	CRT 表示モードの設定
25	カーソル表示の制御
26	カーソル位置の取得
27	アトリビュートの取得
28	スクロール範囲の指定
29	指定された文字のサイズ取得
2A	テキスト画面の初期化
2B	作業領域アドレスの取得
2E	カーソル位置の取得
2F	スプライトスクリーンのライン数変更
30	ソフトキー表示の制御

●呼出し方

- ①レジスタ／メモリにパラメータ設定
- ②AH ← 機能番号
- ③INT 83H

6.4.1 座標系

当 TEXT BIOS における座標系は左上を原点としており、使用時には意味によって次の4種の座標系が存在する。

- ・キャラクタ座標系……………CRT 画面における座標系
- ・スプリットスクリーン座標系……分割画面内における座標系
- ・POPUP スクリーン座標系…………POPUP スクリーン内の座標系
- ・スレームバッファ座標系…………フレームバッファ内の座標系

●キャラクタ座標系

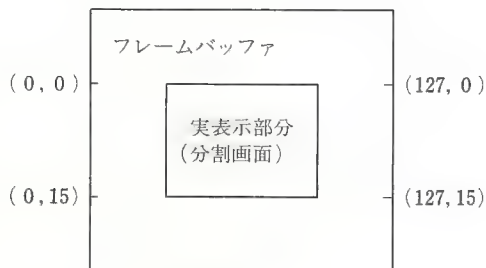
CRT 画面における座標系であり、画面表示文字数分の座標をもつ。たとえば、80×25 の場合には下図のようになる。



●スプリットスクリーン座標系

フレームバッファ内において、Y原点を表示開始行、X原点をフレームバッファ左端とする座標系。

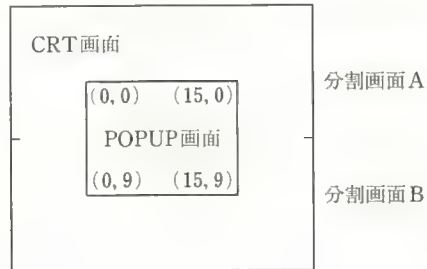
- (例) フレームバッファの幅が 128 文字
分割画面の高さが 16 文字



● POPUP スクリーン座標系

POPUP スクリーンの左上を原点とする座標系，POPUP スクリーンは複数のフレームバッファ(及び分割画面)にまたがることもある。

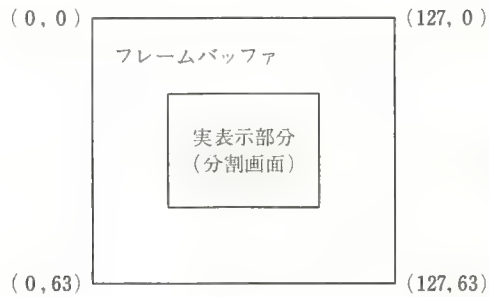
(例) 16×10 の POPUP 画面を開いたとき



● フレームバッファ座標系

フレームバッファの先頭を原点とする座標系。スプリットスクリーン座標系とX座標は同じである。

(例) フレームバッファの幅が 128 文字，高さが 64 文字



6.4.2 ファンクション解説

00H	1 文字表示(コントロールコード有)
-----	--------------------

入 力 AH=00H
DX = 2 バイト文字コード
INT 83H

出 力 なし

解 説 カレントスクリーン(フレームバッファ)に文字を表示する。コントロールコード・エスケープシーケンスは機能する。スクロールやワードラップが許可されているときにはスクロール・ワードラップを行う。フレームバッファの右端1バイトに漢字を表示させるのは、ワードラップが許可されているときに限り、右端にスペシャルスペース (FEH) を置いて次の行にその漢字を置くことによって実現する。

〈使用できるコントロールコード〉

DX=0007H BEL ブザーを鳴らす
0008H BS カーソルを左へ動かす
0009H HT 次のタブストップまでカーソルを移動させる
000AH LF 改行
000CH FF 000AH(LF)と同じ
000DH CR 行の左端までカーソルを動かす
000EH SO G1 のキャラクタセットを選択する
000FH SI G0 のキャラクタセットを選択する
0018H CAN エスケープシーケンスをキャンセルする
001BH ESC エスケープシーケンスに入る

注：SO/SI は状態を保存するだけで、表示文字には影響しない

01H	1 文字表示(コントロールコード無)
-----	--------------------

入 力 AH=01H
DX = 2 バイト文字コード
INT 83H

出 力 なし

解 説 コントロールコード・エスケープシーケンスが特殊なキャラクタとして表示される以外はファンクション 00H と同じ。

02H	ASCIZ 文字列の表示
-----	--------------

入 力 AH=02H
 DX=アトリビュートコード (DH=0)
 DS:SI=混在文字コード文字列の先頭アドレス (00 で終了)
 INT 83H

出 力 CF=1 スクロールが発生した
 =0 スクロールは発生しなかった

解 説 現在のカーソル位置から文字列を表示する。DX で指定されるアトリビュートコードが使用されるが、DX の MSB が 1 であるときには DX の内容ではなく、作業領域のカレントアトリビュートが使用される。
 コントロールコード・エスケープシーケンスは有効であり、スクロール・ワードラップも現在の許可状態に従う。

05H	カレントアトリビュートの設定
-----	----------------

入 力 AH=05H
 DH=00H アトリビュートコード指定を示す
 =01H アトリビュートモード指定を示す
 DL=アトリビュートコード／モード
 INT 83H

出 力 なし

解 説 カレントアトリビュートコードまたはモードを指定する。
 ①アトリビュートコード指定のとき
 カレントアトリビュートコードを更新する。これは文字表示ファンクションで使われる。
 ②アトリビュートモード指定のとき
 カレントアトリビュートモードを更新する。カレントアトリビュートコードは変更されないので注意。アトリビュートモードとしては、0～2のみ有効である。アトリビュートモードについては第4章を参照。

06H	画面へのアトリビュート設定
-----	---------------

入 力 AH=06H
 CX=半角文字数
 DX=アトリビュートコード (DH=0)
 INT 83H

出 力 なし

解 説 現在のカーソル位置から指定された半角文字数分のアトリビュートを設定する。ただし、スペシャルスペース (FEH) はカウントしない。フレームバッファの右端を超えた場合は次の行へ設定するが、表示画面を超えた時点で処理を終了する。
カレントアトリビュートおよびカーソル位置は変化しない。アトリビュートモード 2 における倍角の左右指定、40 桁時の倍角指定は無視される

07H	カラー設定
-----	-------

入 力 AH=07H
DL=上位 4 ビット：バックグラウンドカラー
下位 4 ビット：フォアグラウンドカラー
INT 83H

出 力 なし

解 説 カレントスクリーンの表示色を変更する。ただし、アトリビュートモード 0 時、およびアトリビュートモード 1 時のフォアグラウンドカラーは無視される。これらの場合の表示色は文字単位のアトリビュートの一種として処理する。

08H	カーソル位置の設定
-----	-----------

入 力 AH=08H
DH=X座標(桁指定)スプリットスクリーン座標系
DL=Y座標(行指定)スプリットスクリーン座標系
INT 83H

出 力 なし

解 説 カレントスクリーンのカーソル位置を指定する。指定範囲はファンクション 28H で指定するオリジン/リセットモードによって異なる。オリジンモードの場合には、分割画面内のスクロール範囲の左上を原点とし、スクロール範囲内だけしか指定できず、リセットモードでは分割画面の左上を原点とし、分割画面全体を指定できる。どちらの場合でも分割画面(表示画面)外は指定できない。

09H	カーソル上移動
-----	---------

入 力 AH=09H
CL=行数
INT 83H

出 力 なし

解 説 カレントスクリーンのカーソル位置を指定した行数だけ上移動する。スクロールは行われない。

0AH	カーソル下移動
-----	---------

入 力 AH=0AH
CL=行数
INT 83H

出 力 なし

解 説 カレントスクリーンのカーソル位置を指定した行数だけ下移動する。スクロールは行われない。

0BH	カーソル右移動
-----	---------

入 力 AH=0BH
CL=桁数
INT 83H

出 力 なし

解 説 カレントスクリーンのカーソル位置を指定した桁数(文字数ではない)だけ右移動する。横スクロールが許可されている場合にはスクロールするが、次の行へのラップは行われない。

0CH	カーソル左移動
-----	---------

入 力 AH=0CH
CL=桁数
INT 83H

出 力 なし

解 説 カレントスクリーンのカーソル位置を指定した桁数(文字数ではない)だけ左移動する。横スクロールが許可されている場合にはスクロールするが、次の行へのラップは行われない。

0DH	カーソル下移動(スクロール付)
-----	-----------------

入 力 AH=0DH
INT 83H

出 力 なし

解 説 カレントスクリーンのカーソル位置を1行下げる。カーソルがスクロール範囲の最下行にある場合には、スクロールの許可状態にかかわらず上スクロールが発生する。

0EH	カーソル上移動(スクロール付)
-----	-----------------

入 力 AH=0EH
INT 83H

出 力 なし

解 説 カレントスクリーンのカーソル位置を1行上げる。カーソルがスクロール範囲の最上行にある場合には、スクロールの許可状態にかかわらず下スクロールが発生する。

10H	水平スクロール
-----	---------

入 力 AH=10H
AL=00H 右むき
=01H 左むき
CL=桁数
INT 83H

出 力 CF=0 正常終了
=1 フレームバッファの範囲を越えた
CL=実際にスクロールした桁数

解 説 カレントスクリーン(フレームバッファ)の表示画面を横に移動させる。横スクロールが許可されていない場合にはこのファンクションは無効である。カーソルは表示画面に追従するためカーソルが指す位置は変化しない。

11H	TAB 位置の設定
-----	-----------

入 力 AH=11H
INT 83H

出 力 なし

解 説 現在のカーソル位置の桁にタブストップを設定する。タブストップはすべての分割画面に共通である。デフォルトは8桁おきに設定されている。

12H	TAB 位置の解除
-----	-----------

入 力 AH=12H
AL=00H 現在のカーソル位置を対象とする
=01H すべてのタブストップを対象とする
INT 83H

出 力 なし

解 説 現在のカーソル位置の桁のタブストップ、もしくは全てのタブストップを解除する。

13H	行削除
-----	-----

入 力 AH=13H
CL=行数
INT 83H

出 力 なし

解 説 現在のカーソル行から CL 行を削除し、次の行～表示画面の最下行を削除した部分に移動する。

14H	文字削除
-----	------

入 力 AH=14H
CL=桁数
INT 83H

出 力 なし

解 説 現在のカーソル位置から CL 桁を削除し、次の文字～行の右端を削除した部分に移動する。

15H	行挿入
-----	-----

入 力 AH=15H
CL=行数
INT 83H

出 力 なし

解 説 現在のカーソル行の次の行～表示画面の最下行を CL 行下に移動し、跡を消去する。表示画面を超えた部分は失われる。カーソル位置は変化しない

16H	文字挿入
-----	------

入 力 AH=16H
CL=桁数
INT 83H

出 力 なし

解 説 現在のカーソル位置の次の文字～行の右端を CL 桁右に移動し、跡を消去する。行を超えた部分は失われる。カーソル位置は変化しない。

17H	表示画面のクリア
-----	----------

入 力 AH=17H

AL=00H カーソル位置からスプリットスクリーン座標系の最後まで

=01H スプリットスクリーン座標系の先頭からカーソル位置まで

=02H スプリットスクリーン座標系全て

INT 83H

出 力 なし

解 説 AL で指定された範囲を消去する。カーソル位置やカレントアトリビュートは変化しない。

18H	行のクリア
-----	-------

入 力 AH=18H

AL=00H カーソル位置から行の最後まで

=01H 行の先頭からカーソル位置まで

=02H カーソル行全体

INT 83H

出 力 なし

解 説 AL で指定された範囲を消去する。カーソル位置やカレントアトリビュートは変化しない。

19H	フレームバッファのクリア
-----	--------------

入 力 AH=19H

AL=00H カーソル位置から後に CX ワード

=01H カーソル位置から前に CX ワード

=02H フレームバッファ全体

CX=消去するワード数

INT 83H

出 力 なし

解 説 AL で指定された範囲を消去する。カーソル位置やカレントアトリビュートは変化しない。ラップアラウンドが禁止されている場合、フレームバッファの先頭または最後で消去を終了する。

1AH	フレームバッファ中の位置取得
-----	----------------

入 力 AH=1AH

AL=00H フレームバッファの先頭からのオフセット

=01H TVRAM の先頭からのオフセット

DH=桁位置

DL=行位置

INT 83H

出 力 CF=0 正常終了

=1 エラー

BL=フレームバッファ番号

CX=オフセット

解 説 (DL, DH)で指定された文字のフレームバッファ先頭からのオフセットもしくは TVRAM の先頭からのオフセット(物理アドレス)を計算する。(DL, DH)は(0, 0)から(79, 24)の間である必要がある。

1BH	カレントスクリーンの変更
-----	--------------

入 力 AH=1BH

AL=スクリーン番号

INT 83H

出 力 CF=0 正常終了

=1 指定されたスクリーン番号が存在しない

解 説 カレントスクリーン番号を変更する。

1CH	フレームバッファの定義
-----	-------------

入 力 AH=1CH

CH=設定を開始するフレームバッファブロック番号

CL=設定するフレームバッファ数

ES:BP=設定データへのポインタ

INT 83H

出 力 CF=0 正常終了

=1 異常終了

AH=01H フレームバッファブロック番号が15を超えた

解 説 フレームバッファブロックテーブルを設定する。ブロック番号は0～15の16個である。ひとつのフレームバッファブロックは4バイトで構成されている。

+00	TVRAM 中のフレームバッファアドレス(下位バイト)
+01	TVRAM 中のフレームバッファアドレス(上位バイト)
+02	フレームバッファの幅(偶数)
+03	フレームバッファの高さ

このブロックを設定したい個数だけ用意し、ES:BP でポイントしてこのファンクションを呼ぶ。

CH=n
CL=m
ES:BP

ブロック n
ブロック n+1
ブロック n+2
...
ブロック n+m

1DH 分割画面の設定

入 力 AH=1DH

CL=設定する分割画面数

ES:BP=設定データへのポインタ

INT 83H

出 力 CF=0 正常終了

=1 異常終了

AH=01H 表示画面の高さの合計が画面全体の行数より少ない

AH=02H 分割画面の高さがフレームバッファより大きい

AH=05H システムライン指定が2個以上ある

解 説 フレームバッファから最大4個を CRT 画面に割り付ける。用意された分割画面バッファに従い画面割付けを行う。1つの分割画面バッファは次の通り。

+00	SSATR	0	0	フレームバッファ#			
+01	実画面開始桁位置(フレームバッファ座標系)						
+02	実画面開始行位置(フレームバッファ座標系)						
+03	Reserved						
+04	分割画面の高さ(行数)						
+05	0	0	ATTMODE	0	VROL	HROL	WRAP
+06	バックグラウンドカラー			フォアグラウンドカラー			
+07	Reserved						

SSATR : 分割画面属性

00 ノーマル

01 システムライン

02 未使用

03 未使用

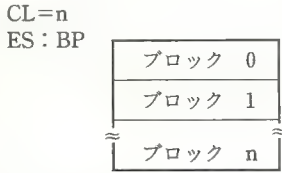
ATTMODE : アトリビュートモード(0~2)

VROL : 縦スクロール制御(1=許可/0=禁止)

HROL : 横スクロール制御(1=許可/0=禁止)

WRAP : ワードラップ(1=許可/0=禁止)

このバッファを設定したい個数だけ用意し、ES:BP でポイントしてこのファンクションを呼ぶ。



1EH	POPUP スクリーンのオープン
-----	------------------

入 力 AH=1EH

AL=スクリーンモード

Bit 2: 縦スクロール(1=許可/0=禁止)

Bit 0: ワードラップ(1=許可/0=禁止)

CH=POPUP スクリーンの幅(4~80)

CL=POPUP スクリーンの高さ(2~25)

DH=POPUP スクリーンの開始桁位置(キャラクタ座標系)

DL=POPUP スクリーンの開始行位置(キャラクタ座標系)

INT 83H

出 力 CF=0 正常終了

=1 エラー

解 説 ポップアップスクリーンを設定する。これ以降、文字表示等の機能はポップアップスクリーンに対して動作する。ポップアップスクリーンで機能するファンクションは、

00H, 01H, 02H, 05H, 06H, 08H, 09H, 0AH, 0BH, 0CH, 0DH, 0EH, 13H, 14H, 15H, 16H, 17H, 18H, 1AH, 26H, 27H

の21種類である。

ポップアップをオープンしたときのアトリビュートモードは必ず1になり、またバックグラウンドカラーは0になる。

1FH	POPUP スクリーンのクローズ
-----	------------------

入 力 AH=1FH

INT 83H

出 力 なし

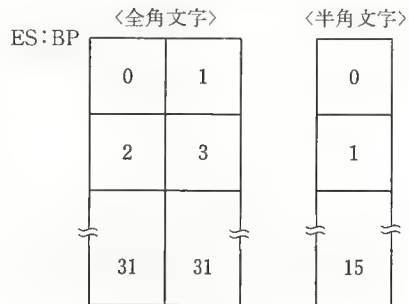
解 説 ポップアップスクリーンを解除し、アトリビュートモード・バックグラウンドカラーを元に戻す。

20H	POPUP スクリーンの画面モード変更
-----	---------------------

入 力 AH=20H
 AL=POPUP スクリーンモード指定
 Bit 2：縦スクロール(1=許可/0=禁止)
 Bit 0：ワードラップ(1=許可/0=禁止)
 INT 83H

出 力 なし

解 説 ポップアップスクリーンの縦スクロール・ワードラップの有無を変更する。



21H	外字フォントの設定
-----	-----------

入 力 AH=21H
 AL=登録するフォントのモード(00H=全角/01H=半角)
 DX=2 バイトコード
 ES:BP=フォントパターンデータの先頭アドレス
 INT 83H

出 力 なし

解 説 RAM キャラクタジェネレータにフォントを登録する(外字登録)。外字には全角と半角とがある。

22H	外字フォントの削除
-----	-----------

入 力 AH=22H
 DX=2 バイトコード
 INT 83H

出 力 なし

解 説 RAM キャラクタジェネレータのフォントを削除する(ブランク文字にする)。

23H 文字フォントの取得

入 力 AH=23H
DX=2 バイトコード
ES:BP=フォントを格納するバッファ
INT 83H

出 力 なし

解 説 キャラクタジェネレータからフォントを読み出す。DX の MSB が 1 の時は、ANK の 8×8 ドットフォントを読む。バッファの大きさは、全角文字のとき 32 バイト、半角文字のとき 16 バイト、8×8 フォントのとき 8 バイトである。

24H CRT 表示モード設定

入 力 AH=24H
AL=CRT 表示モード

MSB	TXON	0	# LINE	0	80CM	INLM	RASM	LSB
-----	------	---	--------	---	------	------	------	-----

TXON : テキスト画面表示制御(1=禁止/0=表示)
LINE : 行数(0=25 行/1=20 行/2=12 行/3=10 行)
80CM : 桁数(0=40 桁/1=80 桁)
INLM : インタレースモード(1=インタレース)
RASM : ラインモード(0=200・400/1=204・408)

INT 83H

出 力 なし

解 説 CRT の表示モードを指定する。

25H カーソル表示の制御

入 力 AH=25H
AL=カーソル指定

MSB	0	0	0	CSTL	CWID	CE	BE	LSB
-----	---	---	---	------	------	----	----	-----

CSTL : カーソル形状
(1=ラインカーソル/0=ブロックカーソル)
CWID : カーソル幅(0=8 ドット/1=16 ドット/2=32 ドット)
CE : カーソル表示(1=ON/0=OFF)
BE : カーソルブリンク(1=ON/0=OFF)

INT 83H

出 力 なし

解 説 カーソルのモードを指定する。

26H	カーソル位置の取得
-----	-----------

入 力 AH=26H
INT 83H

出 力 DH=桁位置(スプリットスクリーン座標系)
DL=行位置(スプリットスクリーン座標系)

解 説 カレントスクリーンの現在のカーソル位置を返す。

27H	アトリビュートの取得
-----	------------

入 力 AH=27H
DH=00H アトリビュートコードを知る
=01H アトリビュートモードを知る
INT 83H

出 力 DX=カレントアトリビュートコード／モード

解 説 カレントスクリーンのカレントアトリビュートモードまたはカレントアトリビュートコードを返す。

28H	スクロール範囲の指定
-----	------------

入 力 AH=28H
AL=オリジンモード(0=リセット／1=セット)
DH=トップマージン(スプリットスクリーンY座標)
DL=ボトムマージン(スプリットスクリーンY座標)
INT 83H

出 力 なし

解 説 カレントスクリーンのスクロール範囲を指定する。スクロール範囲の最小値は2行である。オリジンモードが設定(AL=1)された場合、カーソルはスクロール範囲しか移動せず、スクロール範囲の左上が座標の原点となる。オリジンモードがリセット(AL=0)された場合、カーソルは分割画面全体を移動し、分割画面の左上が座標の原点となる。POP-UP スクリーンに対してはこのファンクションは無効となる。

29H	指定された文字のサイズ取得
-----	---------------

入 力 AH=29H
DX=2バイトコード
INT 83H

出 力 AL=00 全角
=01 半角

解 説 指定されたコードの文字が全角か半角かを判別する。特に外字を判別するときに用いる。

2AH	テキスト画面の初期化
-----	------------

入 力 AH=2AH
INT 83H

出 力 なし

解 説 テキスト画面を初期化する。

2BH	作業領域アドレスの取得
-----	-------------

入 力 AH=2BH
INT 83H

出 力 ES:DX=当 BIOS の作業領域の先頭アドレス

解 説 TEXT BIOS の作業領域アドレスを返す。

2EH	カーソル位置の取得
-----	-----------

入 力 AH=2EH
INT 83H

出 力 DH=桁位置(キャラクタ座標系)
DL=行位置(キャラクタ座標系)

解 説 現在のカーソル位置をキャラクタ座標系で返す。

2FH	ソフトキー表示の制御
-----	------------

入 力 AH=2FH
AL=ファンクションキー表示コントロール

SHMD	0	0	0	ファンクションキー表示個数
------	---	---	---	---------------

SHMD=0:シフトキーを無視する
1:シフト押下でシフト状態の文字列を表示する
ファンクションキー表示個数
0:ファンクションキーを表示しない
5:ファンクションキーを5個表示する
10:ファンクションキーを10個表示する

INT 83H

出 力 CF=0 正常終了
 =1 エラー

解 説 システムラインが設定されていないときはエラーとなる。

30H	スプライトスクリーンのライン数変更
-----	-------------------

入 力 AH=30H
 AL=スプライトモード

SNMD	0	0	0	0	0	0	SPMB
------	---	---	---	---	---	---	------

SNMD=0：TSP に SPRITE ON コマンドのみ設定する

1：同期モードから設定しなおす

SPMB=0：ノーマルモード

1：垂直拡大モード

INT 83H

出 力 なし

解 説 スプライトの垂直拡大の有無を指定する。

6.5 スプライト BIOS

スプライト BIOS は VA のテキスト表示機能を制御する BIOS であり、以下の 24 個の機能がある。

●スプライト BIOS 機能一覧

機能番号 (16 進)	機 能
00	スプライト BIOS の初期化
01	キューの初期化
02	VIEW の定義
03	VIEW の領域確保
04	スプライトの定義
05	スプライト位置指定
06	動作の開始
07	動作の停止
08	表示の開始
09	表示の停止
0A	キューへのデータ入力
0B	キュー内容の初期化
0C	スプライト情報の取得
0D	VIEW 情報の取得
0E	空き領域情報の取得
0F	クリップウィンドウの設定
10	クリップ情報の取得
11	衝突情報の取得
12	クリップ割込みの設定
13	衝突割込みの設定
14	キュー割込みの設定
15	キューバッファデータ量の取得
16	割込み処理のスキップ(スリープ状態)
17	スリープ状態からの取得
18	スリープ情報の取得

●呼出し方

- ① パラメータ設定
- ② AH ← 機能番号
- ③ INT 84H

00H	スプライト BIOS の初期化
-----	-----------------

入 カ AH=00H
AL =垂直解像度
00H=200 ライン
01H=400 ライン
DX=TVRAM の使用開始アドレス
INT 84H

出 力 AL=00H 正常終了
=01H パラメータエラー

解 説 スプライト BIOS を初期化し、同時にスプライトモードを指定する。各ビューデータ／スプライトは次のように初期化される。

ビューデータ：#1～#255 すべてヌルビュー

スプライト：ビュー#0 が1個割り当てられる。座標(0,0)

表示 OFF, 移動停止

01H	キューの初期化
-----	---------

入 力 AH=01H
DX=1 スプライト当りの キューの大きさ
ES:BX=キューの先頭番地
INT 84H

出 力 なし

解 説 スプライトキューを 32 スプライト分用意し、初期化を行う。

02H	VIEW の定義
-----	----------

入 力 AH=02H
ES:BX=ビューパラメータテーブルへのポインタ
INT 84H

出 力 AL=00H 正常終了
=01H メモリ不足

解 説 ビューにパターンを定義する。ビュー番号・パターンのアドレスなどはビューパラメータテーブルで指定する。指定したパターンデータは TVRAM 中の適当な領域に保存される。

<ビューパラメータテーブル>

+00	VNUMBER	DB ? ; ビュー番号(1-255)
+01	VFLAG	DB ? ; 00H=パターン削除/FFH=パタン設定
+02	VTTYPE	DB ? ; 01H=モノクロパターン ; 04H=16色パターン
+03	VCOLOR	DB ? ; ビューカラー(0-31)
+04	VWIDTH	DW ? ; 水平サイズ(8または32の倍数)
+06	VHEIGHT	DW ? ; 垂直サイズ(偶数)
+08	not used	DW ?
+0A	VDATPTR	DD ? ; パターンデータへのポインタ

03H	VIEW の領域確保
-----	------------

入 力 AH=03H

ES: BX=ビューパラメータテーブルへのポインタ

INT 84H

出 力 AL=00 正常終了

=01 メモリ不足

解 説 ビューデータを格納する領域を確保し、そのアドレスを知らせる。

ビューパラメータテーブルの VNUMBER~VHEIGHT をセットし、このファンクションを呼ぶと、VDSIZE(計算された値)・VDATPTR が設定されて戻る。

<ビューパラメータテーブル>

+00 VNUMBER DB ? ; ビュー番号(1-255)

+01 VFLAG DB ? ; 00H=パターン削除/FFH=パターン設定

+02 VTYPE DB ? ; 01H=モノクロパターン

; 04H=16色パターン

+03 VCOLOR DB ? ; ビューカラー(0-31)

+04 VWIDTH DW ? ; 水平サイズ(8 または 32 の倍数)

+06 VHEIGHT DW ? ; 垂直サイズ(偶数)

+08 VDSIZE DW ? ; 返り値

+0A VDATPTR DD ? ; 返り値

04H	スプライトの定義
-----	----------

入 力 AH=04H

AL = スプライト番号(0~31)

CL = ビュー番号の個数

ES: BX=ビュー番号列格納領域(バイト列)

INT 84H

出 力 なし

解 説 指定したスプライトに、ビュー番号列を割り当てる。

05H	スプライト位置指定
-----	-----------

入 力 AH=05H

AL=スプライト番号(0~31)

BL=ビュー番号(0~255)

CX=X座標(-32768~32767)

DX=Y座標(-32768~32767)

INT 84H

出 力 なし

解 説 指定したスプライトにビューを割り当て、指定した座標に置く。

06H	動作の開始
-----	-------

入 力 AH=06H

AL=00H CX:DX で指定されるスプライトを動かす

=01H すべてのスプライトを動かす

DX:DX=DX の MSB:スプライト# 31 のスタート指定

{

CX の LSB:スプライト# 0 のスタート指定

INT 84H

出 力 なし

解 説 指定したスプライトの移動を開始する。

07H	動作の停止
-----	-------

入 力 AH=07H

AL=00H CX:DX で指定されるスプライトを停止する

=01H すべてのスプライトを停止する

DX:DX=DX の MSB:スプライト# 31 の停止指定

{

CX の LSB:スプライト# 0 の停止指定

INT 84H

出 力 なし

解 説 指定したスプライトの移動を停止する。

08H	表示の開始
-----	-------

入 力 AH=08H

AL=00H CX:DX で指定されるスプライトを表示する

=01H すべてのスプライトを表示する

DX:DX=DX の MSB:スプライト# 31 の表示指定

{

CX の LSB:スプライト# 0 の表示指定

INT 84H

出 力 なし

解 説 指定したスプライトの表示を開始する。

09H	表示の停止
-----	-------

入 力 AH=09H

AL=00H CX:DX で指定されるスプライトを表示停止する

=01H すべてのスプライトを表示停止する

DX: CX=DX の MSB: スプライト# 31 の表示停止指定

}

CX の LSB: スプライト# 0 の表示停止指定

INT 84H

出 力 なし

解 説 指定したスプライトの表示を停止する。

0AH	キューへのデータ入力
-----	------------

入 力 AH=0AH

AL=スプライト番号(0-31)

BX=スプライト制御コマンド(0-FH)

CX=パラメータ 1

DX=パラメータ 2

INT 84H

出 力 AL=00H 正常終了

=01H キューに空きがないが、スプライトは MOVE ON の状態

=02H キューに空きがなく、スプライトは MOVE OFF の状態

解 説 スプライト制御コマンドをスプライトキューに入れる。スプライト制御コマンドには 16 種類ある(後述)。このとき 1 または 2 個のパラメータを必要とする。第 1 パラメータを CX に、第 2 パラメータを DX に入れる。

0BH	キュー内容の初期化
-----	-----------

入 力 AH=0BH

AL=スプライト番号(0-31)

INT 84H

出 力 なし

解 説 指定したスプライトキューをクリアする。

0CH	スプライト情報の取得
-----	------------

入 力 AH=0CH

ES: BX=スプライトパラメータテーブルへのポインタ
(SNUMBER にスプライト番号をセットしておく)

INT 84H

出 力 スプライトパラメータテーブル

解 説 指定したスプライトに関する情報をスプライトパラメータテーブルにセットして返す。

<スプライトパラメータテーブル>

SNUMBER	DB ? ; スプライト番号(0~31)
CURVIEW	DB ? ; 現在のビュー番号(0~255)
TONFLAG	DB ? ; 表示フラグ (FFH=ON, 00H=OFF)
MONFLAG	DB ? ; 移動フラグ (FFH=ON, 00H=OFF)
NUMVIEW	DB ? ; ビューデータ列のビューの個数(1~8)
NTHVIEW	DB ? ; ビューデータ列中でのビューの番号(1~8)
VWLIST	DB 8 DUP(?); ビューデータ列
LGXPOS	DB ? ; スプライトの論理水平座標
LGYPPOS	DB ? ; スプライトの論理垂直座標

0DH	VIEW 情報の取得
-----	------------

入 力 AH=0DH

ES: BX=ビューパラメータテーブルへのポインタ
(VNUMBER にビューをセットしておく)

INT 84H

出 力 ビューパラメータテーブル

解 説 指定したスプライトキューをクリアする。

<ビューパラメータテーブル>

+00	VNUMBER	DB ? ; ビュー番号(1-255)
+01	VFLAG	DB ? ; 00H=パターン削除/FFH=パターン設定
+02	VTTYPE	DB ? ; 01H=モノクロパターン ; 04H=16色パターン
+03	VCOLOR	DB ? ; ビューカラー(0-31)
+04	VWIDTH	DW ? ; 水平サイズ(16または32の倍数)
+06	VHEIGHT	DW ? ; 垂直サイズ(偶数)
+08	VDSIZE	DW ? ; パターンデータの大きさ
+0A	VDATPTR	DD ? ; パターンデータへのポインタ

0EH	空き領域情報の取得
-----	-----------

入 力 AH=0EH
 AL=00H TVRAM 前半の空き領域
 =01H TVRAM 後半の空き領域
 INT 84H

出 力 AX=空き領域のサイズ

解 説 TVRAM の空き領域の大きさを返す。

0FH	クリップウィンドウの設定
-----	--------------

入 力 AH=0FH
 ES:BX=クリップウィンドウパラメータテーブルへのポインタ
 INT 84H

出 力 なし

解 説 クリップウィンドウを設定する。

<クリップウィンドウパラメータテーブル>

CLSX1 DW ? ; 左上隅X座標(クリップ左端)

CLSX1 DW ? ; 左上隅Y座標(クリップ上端)

CLSX2 DW ? ; 右下隅X座標(クリップ右端)

CLSX2 DW ? ; 右下隅Y座標(クリップ下端)

10H	クリップ情報の取得
-----	-----------

入 力 AH=10H
 AL=スプライトの選択条件
 00H 全てのスプライト
 01H 表示 ON のスプライト
 02H 表示 ON・移動 ON のスプライト
 ES:BX=スプライト情報テーブル領域へのポインタ
 INT 84H

出 力 クリップ情報のテーブル
 CLST DD ? ; MSB から#31→#0 のスプライトに対応し、クリップ
 ; ウィンドウ外のスプライトに対応するビットが
 ; 1になる。

解 説 クリップウィンドウ外にあるスプライトを知らせる。

11H 衝突情報の取得

入 力 $AH=11H$

AL=スプライトの選択条件

00H 全てのスプライト

01H 表示 ON のスプライト

02H 表示 ON・移動 ON のスプライト

ES:BX=衝突情報テーブル領域へのポインタ

INT 84H

出力 衝突情報テーブル(32 スプライト分)

COST0 DD ?; MSB から# 31 →# 0 のスプライトに対応

COST1 DD ? ; 同 上

§ §

COST31 DD ? ; 同 上

各スプライトに衝突の可能性があるビットが1になる。

解説 衝突の可能性があるスプライトを知らせる。衝突の可能性とは、スプライトの透明部分が重なった時でも衝突とみなすことを示す。

12H クリップ割込みの設定

入 力 $AH=12H$

AL=スプライト番号(0~31)

ES:BX=割込みサービスルーチンアドレス

DX=8000H 割込み設定

=0000H 割込み解除

INT 84H

出力なし

解説 指定したスプライトがクリップウィンドウの内から外へ出たときに実行される割込みサービスルーチンの設定・解除を行う。

13 衝突割込みの設定

入 力 $AH=13H$

ES:BX=割込みのサービスルーチンアドレス

DX=8000H 割込み設定

=0000H 割込み解除

INT 84H

出力なし

解 説 ハードウェアの衝突検出機能により衝突が検出されたときに実行される割込みサービスルーチンの設定・解除を行う。この場合、スプライトの透明な部分が重なっても衝突とはみなさない。

14H	キュー割込みの設定
-----	-----------

入 力 AH=14H
 AL=スプライト番号(0~31)
 ES:BX=割込みサービスルーチンアドレス
 DX=MSB:1 割込み設定
 :0 割込み解除
 bit 4-0: 切込みを起こすデータ量(バイト)
 INT 84H

出 力 なし

解 説 スプライトキュー内のデータ量が、指定した値以下になったときに実行される割込みサービスルーチンの設定・解除を行う。

15H	キューバッファデータ量の取得
-----	----------------

入 力 AH=15H
 AL=スプライト番号(0~31)
 INT 84H

出 力 AX=スプライトキュー内のデータ量(バイト)

解 説 スプライトキュー内のデータ量を返す。

16H	割込み処理のスキップ(スリープ状態)
-----	--------------------

入 力 AH=16H
 INT 84H

出 力 なし

解 説 スプライト BIOS の割込み処理を停止し、システム全体の速度を向上させる。

17H	スリープ状態からの復帰
-----	-------------

入 力 AH=17H
 INT 84H

出 力 なし

解 説 SLEEP 状態から復帰する。

18H	スリープ情報の取得
-----	-----------

入 力 AH=18H
INT 84H

出 力 AL=00H : WAKE UP 状態
FFH : SLEEP 状態
AH=MSB : 1=スプライト垂直拡大表示, 0=ノーマル表示
LSB : 1=400 ラインモード, 0=200 ラインモード

解 説 SLEEP/WAKE UP 状態及びスプライト垂直解像度状態を返す。

スプライト制御コマンド(ファンクション 0AH 参照)

00H	[パラメータ 1]で指定した値だけ上に移動させる
01H	[パラメータ 1]で指定した値だけ下に移動させる
02H	[パラメータ 1]で指定した値だけ左に移動させる
03H	[パラメータ 1]で指定した値だけ右に移動させる
04H	[パラメータ 1]で指定した時間(秒)だけ待つ
05H	移動速度を[パラメータ 1]ピクセル/秒にする
06H	移動速度を現在よりも[パラメータ 1]ピクセル/秒変化させる
07H	ビュー番号を変化させる時間を[パラメータ 1]で指定(1/60 秒単位)
08H	ビュー番号を変化させる時間を[パラメータ 1]だけ変更(1/60 秒単位)
09H	ビュー番号を, ビュー番号列の[パラメータ 1]番目に変更する
0AH	ビュー番号を, [パラメータ 1]に変更する
0BH	別のスプライト([パラメータ 1]=スプライト番号)の位置へ移動する
0CH	位置([パラメータ 1], [パラメータ 2])にジャンプさせる
0DH	位置を([パラメータ 1], [パラメータ 2])だけジャンプさせる
0EH	位置を座標([パラメータ 1], [パラメータ 2])に移動させる
0FH	位置を([パラメータ 1], [パラメータ 2])だけ移動させる

6.6 グラフィック画面制御 BIOS

グラフィック画面制御 BIOS は VA のグラフィック画面の表示モード・フレームバッファ・パレットなどを制御する BIOS であり、以下の 12 個の機能がある。

●グラフィック画面制御 BIOS 機能一覧

機能番号 (16 進)	機 能
00	画面モードの設定
01	フレームバッファの定義
02	ウィンドウの定義
03	画面合成の設定
04	表示位置の設定
05	表示位置の移動
06	画面マスクの設定
07	フレームバッファ情報の取得
08	パレットの設定
09	パレットモードの設定
0A	パレットの初期化
0B	グラフィック表示の ON/OFF

●呼出し方

- ① パラメータ設定
- ② AH ← 機能番号
- ③ INT 8FH

レジスタは AX 以外は値を返す場合を除いて保存される。
また、AX にはリターンコードが設定される。

●グラフィック画面制御 BIOS リターンコード一覧

0	正常終了
1	未定義フレームバッファを参照している
2	ピクセルサイズが間違っている
3	フレームバッファ番号が間違っている
4	表示位置が間違っている
5	フレームバッファの大きさが間違っている
6	VRAM が不足している
7	ページ番号が間違っている
8	指定したフレームバッファは表示されていない
9	ウィンドウが間違っている

00H	画面モードの設定
-----	----------

入 力 AH=00H

BX=画面モード

bit 15: 1=シングルプレーン / 0=マルチプレーン

bit 14: 1=2画面モード / 0=1画面モード

bit 13: 1=グラフィック表示 ON / 0=OFF

bit 3: グラフィック画面 0 水平解像度 (1=320 / 0=640)

bit 2: グラフィック画面 1 水平解像度 (1=320 / 0=640)

bit 1-0: グラフィック画面垂直解像度

(00=400line / 01=408line)

(10=200line / 11=204line)

CL=グラフィック画面 0 ピクセルサイズ (1, 3, 4, 5, 8, 16 のうち 1 つ)

CH=グラフィック画面 1 ピクセルサイズ (1, 3, 4, 5, 8, 16 のうち 1 つ)

DH=フォアグラウンドカラー (シングルプレーンモノクロのとき)

DL=マルチプレーンモノクロ時の表示プレーン

bit 3: プレーン 3 表示制御 (1=ON / 0=OFF)

bit 2: プレーン 2 表示制御 (1=ON / 0=OFF)

bit 1: プレーン 1 表示制御 (1=ON / 0=OFF)

bit 0: プレーン 0 表示制御 (1=ON / 0=OFF)

INT 8FH

出 力 AX=リターンコード

解 説 画面モードを設定し、フレームバッファ・ウィンドウ・画面合成を初期状態に戻す。2画面モード時のスクリーン 1 に関して V3 BASIC の規定と同様の制限がある。

01H	フレームバッファの定義
-----	-------------

入 力 AH=01H

AL=グラフィック画面番号 (0 または 1)

CX=定義するバッファ数

ES: DI=バッファディスクリプタリストのアドレス

INT 8FH

出 力 AX=リターンコード

解 説 VRAM 中にフレームバッファを定義する。スクリーン 0 / 1 別々に定義を行う。ここで定義されたフレームバッファにはディスクリプタリスト順にバッファ番号 (0 ~) が付与され、以後フレームバッファの参照にはバッファ番号を使用する。グラフィック画面 0 と 1 は別々に管理される。

<バッファディスクリプタリスト>

各6バイトからなるバッファディスクリプタを、定義する数だけ連続して並べたもの。
各バッファディスクリプタの内容は次のとおり。

dw ピクセルサイズ(1,4,8,16のどれか)

dw バッファの幅(ピクセル)

dw バッファの高さ(ピクセル)

ピクセルサイズは1,4,8,16のどれかを使用する。32色モードの場合は8を、8色モードの場合は4を使用する。

02H	ウィンドウの定義
-----	----------

入 力 AH=02H

AL=グラフィック画面番号(0または1)

CX=定義するウィンドウ数

ES:DI=ウィンドウディスクリプタリストのアドレス

INT 8FH

出 力 AX=リターンコード

解 説 グラフィック画面の画面分割およびフレームバッファの実画面への対応を V3 BASIC の規定と同様に定義する。

<ウィンドウディスクリプタリスト>

各10バイトからなるウィンドウディスクリプタを、定義する数だけ連続して並べたもの。各ウィンドウディスクリプタの内容は次の通り。

dw この分割画面に表示するフレームバッファ番号

dw 分割画面の垂直位置

dw 分割画面の高さ

dw フレームバッファ中の表示開始位置のX座標

dw フレームバッファ中の表示開始位置のY座標

03H	画面合成の設定
-----	---------

入 力 AH=03H

AL=テキスト/スプライト判別境界色

CX=合成順位

bit 0-3: 最高優先順位の画像指定

bit 4-7: 第2優先順位の画像指定

bit 8-11: 第3優先順位の画像指定

bit 12-15: 最低優先順位の画像指定

各4ビットには0~4の値を設定する。

0: 画像なし

1: テキスト画面

- 2: スプライト画面
- 3: グラフィック画面 0
- 4: グラフィック画面 1

INT 8FH

出 力 AX=リターンコード

解 説 画面合成の順位を設定する。V3 BASIC の COMPOSE 文に相当する。

04H	表示位置の設定(絶対位置)
-----	---------------

入 力 AH=04H

AL=グラフィック画面番号(0 または 1)

CL=バッファ番号

BX=フレームバッファ中の表示開始位置X座標

DX=フレームバッファ中の表示開始位置Y座標

INT 8FH

出 力 AX=リターンコード

解 説 表示画面のフレームバッファ中の位置を移動する。V3 BASIC の ROLL TO に相当する。

05H	表示位置の移動(相対位置)
-----	---------------

入 力 AH=05H

AL=グラフィック画面番号(0 または 1)

CL=バッファ番号

BX=表示開始位置の水平方向移動量

DX=表示開始位置の垂直方向移動量

INT 8FH

出 力 AX=リターンコード

解 説 表示画面のフレームバッファ中の位置を移動する。V3 BASIC の ROLL 文に相当する。

06H	画面マスクの設定
-----	----------

入 力 AH=06H

AL=マスクモード

CL=マスクの挿入位置(マスクスロット番号)

BX=画面マスクの左端(0-639)

DX=画面マスクの上端(0-199)

SI=画面マスクの右端(0-639 で BX 以上)

DI=画面マスクの下端(0-199 で DX 以上)

INT 8FH

出 力 AX=リターンコード

解 説 画面マスクを設定する。V3 BASIC の SCREEN MASK 文に相当する。

07H	フレームバッファ情報の取得
-----	---------------

入 力 AH=07H

AL=グラフィック画面番号(0または1)

CL=バッファ番号

INT 8FH

出 力 AX=リターンコード

ES:DI=フレームバッファ情報テーブルアドレス(正常終了時)

<フレームバッファ情報テーブル>

dw ピクセルサイズ

dw フレームバッファの幅(ピクセル)

dw フレームバッファの高さ(ピクセル)

dw フレームバッファのアドレス(下位 16 ビット)

dw フレームバッファのアドレス(上位 16 ビット)

解 説 定義されているフレームバッファの情報を得る。フレームバッファの実アドレスを知ることができる。

08H	パレットの設定
-----	---------

入 力 AH=08H

AL=パレット番号(0~32)

CX=パレットの値

INT 8FH

出 力 AX=リターンコード

解 説 パレットを変更する。2組あるカラーパレットを連続して扱い、カラーパレット0はパレット番号0~15で、カラーパレット1は番号16~31で選択する。このファンクションを利用しないでパレットレジスタをポートから直接変更した場合、作業領域が更新されないため画面コピー時に異なった色で出力される。

09H	パレットモードの設定
-----	------------

入 力 AH=09H

AL=パレットモード

bit 7,6: パレットモード (0~3)

bit 5,4: パレットモード 2 のときの画像指定

(0=テキスト, 1=スプライト, 2=グラフィックス, 3=グラフィックス 2)

bit 3,2: プリンクモード (プリンクサイクル)

bit 1,0: プリンク比率

INT 8FH

出 力 AX=リターンコード

解 説 パレットモードを変更する。V3 BASIC の PALLETE MODE 文に相当する。

0AH	パレット初期化
-----	---------

入 力 AH=0AH

INT 8FH

出 力 AX=リターンコード

解 説 パレットの値を初期化する。V3 BASIC のパラメータなし PALLETE 文に相当する。

0BH	グラフィック表示の ON / OFF
-----	--------------------

入 力 AH=0BH

AL=00 全てのグラフィック表示を停止する。

01 グラフィック表示を再開する。マルチプレーンモノクロモード時の表示プレーンの指定が DL レジスタでできる。

DL=表示プレーン指定 (ファンクション 0 と同じ)

INT 8FH

出 力 AX=リターンコード

解 説 グラフィック画面の表示 ON/OFF を行う。

作業領域 0338H : 0000H～

グラフィック画面制御 BIOS の作業領域はセグメント 338H である。

OFFSET (16 進)	
00-0E 0F-10	内部使用 画面モード bit 15 : 1=シングルプレーンモード / 0=マルチプレーンモード bit 14 : 1=2 画面モード / 0=1 画面モード bit 13 : 1=表示 ON / 0=表示 OFF bit 4 : グラフィック画面 0 水平解像度 (1=320, 0=640) bit 3 : グラフィック画面 1 水平解像度 (1=320, 0=640) bit 2 : 垂直解像度 (1=200 / 204, 0=400 / 408)
11	グラフィック画面 0 のピクセルサイズ
12	グラフィック画面 1 のピクセルサイズ
13	グラフィック画面 0 用のフレームバッファ数
14	グラフィック画面 1 用のフレームバッファ数

<グラフィック画面 0 用フレームバッファ情報テーブル>

17-20	フレームバッファ# 0 情報テーブル
21-2A	フレームバッファ# 1 情報テーブル
2B-34	フレームバッファ# 2 情報テーブル
35-3E	フレームバッファ# 3 情報テーブル
3F-48	フレームバッファ# 4 情報テーブル
49-52	フレームバッファ# 5 情報テーブル
53-5C	フレームバッファ# 6 情報テーブル
5D-66	フレームバッファ# 7 情報テーブル

<グラフィック画面 1 用フレームバッファ情報テーブル>

67-70	フレームバッファ# 0 情報テーブル
71-7A	フレームバッファ# 1 情報テーブル
7B-84	フレームバッファ# 2 情報テーブル
85-8E	フレームバッファ# 3 情報テーブル
8F-98	フレームバッファ# 4 情報テーブル
99-A2	フレームバッファ# 5 情報テーブル
A3-AC	フレームバッファ# 6 情報テーブル
AD-B6	フレームバッファ# 7 情報テーブル

<ウィンドウディスクリプタテーブル>

B7-C0	グラフィック画面 0 ウィンドウ 0 のディスクリプタ
C1-CA	グラフィック画面 0 ウィンドウ 1 のディスクリプタ
CB-D4	グラフィック画面 0 ウィンドウ 2 のディスクリプタ
D5-DE	グラフィック画面 1 ウィンドウのディスクリプタ

6.7 拡張グラフィック BIOS

拡張グラフィック BIOS は VA のグラフィック画面への描画機能をサポートする BIOS であり、25 個の機能がある。

●拡張グラフィック BIOS 機能一覧

機能番号 (16 進)	機 能
01	アクティブフレームバッファの設定
02	ビューの設定
10	ビューポート内の消去(塗りつぶし)
11	ライン描画
12	多角形描画
13	多ライン描画
14	円描画
15	ペイント
16	ドット描画
17	ポイント読取り
18	パターンゲット
19	パターンプット
31	描画面と表示画面の切換え
32	グラフィック画面への文字列書込み
33	グラフィック画面への文字書込み
34	仮想画面の設定
35	仮想画面の表示
37	パターン拡大・縮小プット
3E	グラフィック画面拡大・縮小
38	3D→2D変換
39	図形の回転
3A	画面圧縮セーブ
3B	画面再生ロード
3C	色の一括変更
3D	ピクセルサイズ 16→8 変換

●呼出し方

- ①レジスタ／メモリにパラメータ設定
- ②AH ← 機能番号
- ③INT 87H

01H	アクティブフレームバッファの設定
-----	------------------

入 力 AH=01H
ES:DX=フレームバッファ情報アドレス
INT 87H

出 力 なし

解 説 アクティブフレームバッファを設定する。以後このフレームバッファに対して描画が行われる。VIEW 等は設定されない。

<フレームバッファ情報>

- DW 画面モード
- DW フレームバッファオフセットアドレス
- DW フレームバッファセグメントアドレス
- DW フレームバッファ X 方向サイズ(バイト単位)
- DW フレームバッファ Y 方向サイズ

<画面モード>

- 0 マルチプレーン 4 バンク
- 1 シングルプレーン 1 ビット
- 2 シングルプレーン 4 ビット
- 3 シングルプレーン 8 ビット
- 4 シングルプレーン 16 ビット
- 8 マルチプレーン 1 バンク(プレーン 0)
- 9 マルチプレーン 1 バンク(プレーン 1)
- 10 マルチプレーン 1 バンク(プレーン 2)
- 11 マルチプレーン 1 バンク(プレーン 3)
- 12 マルチプレーン 3 バンク

02H	ビューの設定
-----	--------

入 力 AH=02H
AL=動作モード
ES:BX=VIEW データテーブルアドレス
INT 87H

出 力 AX=0:正常終了
=1:VIEW 値が間違っている
=2:VIEW で指定した領域が、フレームバッファよりも広い

解説 VIEW ポートと描画原点をセットする。

<動作モード>

- bit 1:0=描画時に VIEW でクリッピングを行う
- 1=描画時に VIEW でクリッピングを行わない

bit 0:0=原点移動の処理を行う
1=原点移動の処理を行わない

<VIEW データテーブル>

DW VIEW 領域の右端
DW VIEW 領域の上端
DW VIEW 領域の左端
DW VIEW 領域の下端
DW 描画原点の X 座標
DW 描画原点の Y 座標

10H	ビューポート内の消去(塗りつぶし)
-----	-------------------

入 力 AH=10H
CX=塗りつぶす色
INT 87H

出 力 なし

解 説 ビューポート内を指定した色で塗りつぶす。

11H	ライン描画
-----	-------

入 力 AH=11H
AL=コントロールデータ
ES:BX=ラインテーブルアドレス
CX:DX=タイルテーブルアドレス
INT 87H

出 力 なし

解 説 直線、長方形などを描く。XOR, OR, AND 等の指定・ラインスタイル指定・BOX 内のタイル FILL・枠線付き BOX FILL などの指定が可能。

<コントロールデータ>

bit 7:1=タイル指定あり
0=タイル指定なし
bit 6:1=BOX 指定時 FILL する
0=BOX 指定時 FILL しない
bit 5:1=BOX
0=LINE
bit 4:1=FILL 指定時枠線あり
0=FILL 指定時枠線なし
bit 3:0

bit 2-0: 条件

- 000 PSET
- 001 XOR
- 010 AND
- 011 CLEAR
- 100 PRESET
- 101 OR

<ラインテーブル>

- DW 始点 X, 始点 Y
- DW 終点 X, 終点 Y
- DW ラインスタイル
- DW カラー 1 (BOX の枠)
- DW カラー 2 (BOX の内部)

<タイルテーブル> 必ず偶数番地から始まること

○マルチプレーン 4 ビット/ピクセル

- DW タイルパターンの縦の長さ (最大 255)
- DW B, R, G, I ; 1 ライン目
- {

- DW B, R, G, I ; n ライン目

○マルチプレーン 3 ビット/ピクセル

- DW タイルパターンの縦の長さ (最大 255)
- DW B, R, G, I (ただし, I はダミーパラメータ)
- {

- DW B, R, G, I ; n ライン目

○マルチプレーン 1 ビットピクセルモード

- DW タイルパターンの縦の長さ (最大 255)
- DW D, D, D, D (すべて同じデータ)
- {

- DW B, R, G, I ; n ライン目

○シングルプレーンモード (GET データと同形式)

- DW 横ピクセル数, 縦ピクセル数
- DW データ...

12H	多角形描画
-----	-------

入 力 AH=12H

AL=コントロールデータ

ES:BX =ラインテーブルアドレス

CX:DX =タイルテーブルアドレス

INT 87H

出 力 なし

解 説 多角形を描く。スタイルは LINE と同じ。三角形、四角形の場合はタイルによる塗りつぶし可。

<コントロールデータ>

bit 7:1=タイル指定あり
 0=タイル指定なし
 bit 6:1=FILL(三角形・四角形時のみ)
 0=FILL しない
 bit 5:0
 bit 4:1=FILL 指定時枠線あり
 0=FILL 指定時枠線なし
 bit 3:0
 bit 2-0:条件
 000 PSET
 001 XOR
 010 AND
 011 CLEAR
 100 PRESET
 101 OR

<ラインテーブル>

DW ラインデータオフセットアドレス
 DW ラインデータセグメントアドレス
 DW ラインの本数(n)
 DW 0
 DW ラインスタイル
 DW カラー 1 (枠)
 DW カラー 2 (内部)

<ラインデータ>

DW X 1, Y 1
 DW X 2, Y 2
 }
 DW Xn, Yn

13H	多ライン描画
-----	--------

入 力 AH=13H
 AL=コントロールデータ
 ES:BX=ラインテーブルアドレス
 INT 87H

出 力 なし

解 説 多数の線を一度に書く。

<コントロールデータ>

bit 7:1=色コードあり(各線に色指定する)
 0=色コードなし
 bit 6-3:0000
 bit 2-0:条件
 000 PSET
 001 XOR
 010 AND
 011 CLEAR
 100 PRESET
 101 OR

<ラインテーブル>

DW ラインデータオフセットアドレス
 DW ラインデータセグメントアドレス
 DW ラインの本数(n)
 DW 0
 DW ラインスタイル
 DW カラー

<ラインデータ>

DW AX1, AY1
 DW AX2, AX2(, COLOR)
 DW BX1, BY1
 DW BX2, BY2(, COLOR)
 {
 DW nX1, nY1
 DW nX2, nY2(, COLOR)

14H	円描画
-----	-----

入 力 AH=14 H

AL=コントロールデータ

ES: BX=ラインテーブルアドレス

CX: DX=タイルテーブルアドレス

INT 87 H

出 力 なし

解 説 円・円弧・扇を書く。タイルによる塗りつぶし可。開始点と終了点が一致した場合は1点のみを描画する。開始点または終了点が円と一致しない場合は一番近い値になる。

<コントロールデータ>

bit 7:1=タイル指定あり

0=タイル指定なし

bit 6:1=FILL する

0=FILL しない

bit 5:1=終了点半径描く

0=終了点半径描かない

bit 4:1=開始点半径描く or FILL 指定時枠線あり

0=開始点半径描かない

or FILL 指定時枠線なし

bit 3:1=開始点終了点指定あり

0=開始点終了点指定なし

bit 2-0: 条件

000 PSET

001 XOR

010 AND

011 CLEAR

100 PRESET

101 OR

<サークルデータテーブル>

DW	X, Y	; 中心点座標
DW	RX, RY	; X, Y 各半径
DW	0	
DW	カラー 1	; 描画色
DW	カラー 2	; 枠線色
DW	開始点 X, 開始点 Y	; 描画開始座標
DW	終了点 X, 終了点 Y	; 描画終了座標

15H	ペイント
-----	------

入 力 AH=15 H

AL=コントロールデータ

ES: BX=ペイントデータテーブルアドレス

CX: DX=タイルテーブルアドレス

INT 87 H

出 力 AX=0: 正常終了

=1: STOP が押下のため中断した

=2: ワークエリア不足のため中断した

解 説 境界色で囲まれた範囲を塗りつぶす。

<コントロールデータ>

bit 7: 1=タイル指定あり

0=タイル指定なし

bit 6-3: 0000

bit 2-0: 条件

000 PSET

001 XOR

010 AND

011 CLEAR

100 PRESET

101 OR

<ペイントデータテーブル>

DW 開始座標 X, 開始座標 Y

DW 塗りつぶす色

DW 境界色

DW ワークエリアオフセット

DW ワークエリアセグメント

DW ワークエリアの終了オフセット

16H	ドット描画
-----	-------

入 力 AH=16 H

DX=X 座標

BX=Y 座標

CX=カラー

AL=オペレーション

0 PSET

1 XOR

- 2 AND
- 3 CLEAR
- 4 PRESET
- 5 OR

INT 87 H

出 力 AX=0：正常終了
 =1：座標が VIEW の外だった

解 説 指定された座標にドットを描く。

17H	ポイント読取り
-----	---------

入 力 AH=17H
 DX=X 座標
 BX=Y 座標
 INT 87H

出 力 AX=0：正常終了
 =1：座標が VIEW の外だった
 CX=カラー(正常終了時)

解 説 指定された座標のドットの色を読み取る。

18H	パターンゲット
-----	---------

入 力 AH=18H
 ES:BX=データテーブルアドレス
 INT 87H

出 力 AX=0：正常終了
 =1：座標が VIEW の外だった
 =2：格納領域のサイズが小さ過ぎる
 =3：データテーブルに無効なパラメータがある

解 説 パターンの GET を行う。

＜データテーブル＞

- DW 開始座標 X
- DW 開始座標 Y
- DW 水平サイズ
- DW 垂直サイズ
- DW 格納領域のサイズ
- DW 格納領域のオフセット
- DW 格納領域のセグメント

19H	パターンプット
-----	---------

入 力 AH=19H

AL=コントロールデータ

ES:BX=データテーブルアドレス

INT 87H

出 力 AX=0:正常終了

=1:座標が VIEW の外だった

=2:格納領域のサイズが小さ過ぎる

=3:データテーブルに無効なパラメータがある

解 説 パターンの PUT を行う。

<コントロールデータ>

bit 7-6:10=デスティネーションが 0 の部分だけ転送

01=ソースが 0 の部分は転送しない

00=通常の PUT

bit 5:1=1 プレーンからのピクセルサイズ拡張 PUT

0=同じピクセルサイズで PUT

bit 4:1=パターン Blt (PatBlt)

0=通常の Blt (BitBlt)

bit 3-0:条件(S=PUT するパターン D=PUT される画面)

0:PSET [S] 8:NOT(S OR D)

1:XOR [S XOR D] 9:NOT(S XOR D)

2:AND [S AND D] A:NOT(S) AND D

3:CLEAR [0] B:NOT(S) OR D

4:PRESET [NOT(S)] C:NOT(D)

5:OR [S OR D] D:S OR NOT(D)

6:S AND NOT(D) E:NOT(S AND D)

7:NOP F:1

<データテーブル>

DW 開始座標 X

DW 開始座標 Y

DW 水平サイズ (PatBlt 時のみ)

DW 垂直サイズ (PatBlt 時のみ)

DW 格納領域のサイズ

DW 格納領域のオフセット

DW 格納領域のセグメント

DW フォアグラウンドカラー

DW バックグラウンドカラー

31H	描画面と表示画面の切換え
-----	--------------

入 力 AH=31H

AL=モード

CL=ページ番号(モード0のとき)

ES:BX=フレームバッファテーブルアドレス(モード0のとき)

INT 87H

出 力 なし

解 説 BIOS ファンクションにおいての描画面と表示画面を切り換え、表示が終了した画面を消去する。これによって、アプリケーション側は、表示ページを気にすることなく、1つの画面に連続して書き込むだけで、アニメーションが簡単に実現する。

<モード>

- 0 イニシャライズ(フレームバッファの定義)
- 1 画面1を表示し、画面2をアクティブにし、CLSする
- 2 画面2を表示し、画面1をアクティブにし、CLSする
- 3 表示画面とアクティブ画面を入れ換え、アクティブ画面をCLSする

<フレームバッファテーブル>

- DW フレームバッファの数 (注1)
- DW フレームバッファテーブル (注2)
- DW 画面1のウィンドウ数 (注3)
- DW 画面1のウィンドウテーブル(注4)
- DW 画面1のフレームバッファオフセット
- DW 画面1のフレームバッファセグメント
- DW 画面2のウィンドウ数
- DW 画面2のウィンドウテーブル
- DW 画面2のフレームバッファオフセット
- DW 画面2のフレームバッファセグメント

注1: グラフィック画面制御 BIOS のファンクション 01(フレームバッファの定義)の CX と同じ

注2: グラフィック画面制御 BIOS のファンクション 01(フレームバッファの定義)の DI と同じ

注3: グラフィック画面制御 BIOS のファンクション 02(ウィンドウの定義)の CX と同じ

注4: グラフィック画面制御 BIOS のファンクション 02(ウィンドウの定義)の DI と同じ

32H	グラフィック画面への文字列書込み
-----	------------------

入 力 AH=32H

DS: BX=ストリングデータテーブルアドレス

ES: DX=フォントデータテーブルアドレス

INT 87H

出 力 なし

解 説 グラフィックパターンで定義されている文字フォントにより、文字列をグラフィック画面に表示する。マルチプレーンモードの場合、フォントデータのバンク数は任意(1-4)で、各バンクを実際に表示するプレーンに自由に割り当てることができる。また、ドットパターンを反転して表示することもできる。シングルプレーンモード時に、フォントデータが1ビット/ピクセルの場合、フォアグラウンド・バックグラウンドカラーを指定する。表示する文字列はフレームバッファの右端を越えないようにしなければならない。文字列は、0を終了コードをする ASCIZ 文字列を使用する。フォントデータのサイズは64 K バイト以上も可能である。

<ストリングデータテーブル>

DW 表示開始 X 座標(バイトバウンダリの位置)

DW 表示開始 Y 座標

DB ASCIZ 文字列

<フォントデータテーブル>

○マルチプレーンモード

DW 描画プレーン情報

bit 15-12: プレーン 3

bit 11- 8: プレーン 2

bit 7- 4: プレーン 1

bit 3- 0: プレーン 0

各プレーンの bit 3-2: 使用するフォントのバンク番号

bit 1: 1=反転/0=ノーマル

bit 0: 1=描画する/0=描画しない

DB フォントデータのバンク数-1

DB フォントサイズ

bit 5-4: Y ドットサイズ(00=8/01=16/10=24/11=32)

bit 1-0: X ドットサイズ(00=8/01=16/10=24/11=32)

DW フォントデータオフセット

DW フォントデータセグメント

●フォントデータ

バンク 0:

DB X0-7, X8-15, ...; 1 ラスタ目

DB X0-7, X8-15, ...; 2 ラスタ目

DB X0-7, X8-15, ...; 3 ラスタ目


```

    }
    DB    X0-7, X8-15, ...;最終ラスタ
バンク1:
    DB    X0-7, X8-15, ...; 1 ラスタ目
    DB    X0-7, X8-15, ...; 2 ラスタ目
    }
最終バンク:
    DB    X0-7, X8-15, ...; 1 ラスタ目
    DB    X0-7, X8-15, ...; 2 ラスタ目
    }

```

○シングルプレーンモード

```

DW    フォントデータのピクセルサイズ(1, 4, 8, 16)
DW    フォント X サイズ(ドット)
DW    フォント Y サイズ(ドット)
DW    フォントデータオフセット
DW    フォントデータセグメント
DW    フォアグラウンドカラー
DW    バックグラウンドカラー

```

●フォントデータ

```

DW    X0-15, X16-32, ...; 1 ラスタ目
DW    X0-15, X16-32, ...; 2 ラスタ目
DW    X0-15, X16-32, ...; 3 ラスタ目
    }
DW    X0-15, X16-32, ...;最終ラスタ

```

33H	グラフィック画面への文字書込み
-----	-----------------

入 力 AH=33H

SI:DI=描画する画面のアドレス(マルチプレーンモード)

描画する画面の座標(シングルプレーンモード)

CX=文字コード

ES:DX=フォントデータテーブルアドレス

(ファンクション 32 と同じ)

INT 87H

出 力 なし

解 説 ファンクション 32 と同様にグラフィック画面に 1 文字表示する。

34H	仮想画面の設定
-----	---------

入 力 AH=34H
 ES:DX=マップワークアドレス
 INT 87H

出 力 なし

解 説 メインメモリに仮想画面を設定する。仮想画面は任意の大きさを持つことができ、各バイトがキャラクタ番号を表す。このキャラクタ番号のグラフィックフォントパターンを実画面に転送する(ファンクション 35)ことにより、ウィンドウ機能を実現する。実際に表示されるのは仮想画面の一部だけであり、表示領域を変更することにより大きな画面の中を動きまわっている様に見える。また、仮想画面を複数設定し、表示画面の窓に対応させることにより、マルチウィンドウ機能が実現できる。なお、フォントテーブルの形式はファンクション 32/33 と同じである。

<ES:DX マップワーク>

16 バイト	フォントデータテーブル(ファンクション 32 と同じ)
32 バイト	システム用作業領域
DW	実画面の表示開始座標 X
DW	実画面の表示開始座標 Y
DW	実画面の大きさ X(キャラクタ単位)
DW	実画面の大きさ Y(キャラクタ単位)
DW	仮想画面の大きさ X(キャラクタ単位)
DW	仮想画面の大きさ Y(キャラクタ単位)
DW	仮想画面開始アドレスのオフセット
DW	仮想画面作業領域開始アドレスのオフセット

仮想画面のサイズ=幅×高さ×2(バイト)

仮想画面作業領域のサイズ=幅×高さ×2(バイト)

35H	仮想画面の表示
-----	---------

入 力 AH=35H
 AL=全転送フラグ(1=全転送)
 SI=仮想画面中の表示開始 X 座標
 DI=仮想画面中の表示開始 Y 座標
 ES:DX=マップワークアドレス
 INT 87H

出 力 なし

解 説 ファンクション 34 で設定した仮想画面を実画面に転送する。転送される実画面の大きさは仮想画面の 1 キャラクタ当りの大きさによって左右される。全転送フラグが 0 以外のときは、転送領域の全キャラクタを転送し、0 のときは、前回呼ばれたときから変化したキャラクタのみ転送する。

37H	パターン拡大・縮小
-----	-----------

入 力 AH=37H
 AL=0:2倍拡大
 1:1/2縮小
 ES:BX=転送テーブルアドレス
 INT 87H

出 力 なし

解 説 GET データを拡大／縮小してコピーする

<転送座標テーブルアドレス>

DD ソース GET データテーブルアドレス
 DD デスティネーション GET データテーブルアドレス

38H	3D → 2D 変換
-----	------------

入力 AH=38H
 AL=00H:色コードなし
 80H:色コードあり
 CX=ライン総数
 DX=視点距離
 BX=パース
 DS:SI=3次元ラインテーブルアドレス
 ES:DI=2次元ラインテーブルアドレス
 INT 87H

出 力 CX=二次元ラインの本数

解 説 3次元データを2次元上に投影する。データを、 $X=Z$, $-X=Z$, $Y=Z$, $-Y=Z$ の視野ピラミッドでクリッピングし、次の式で計算して2次元画面に投影する。

$$X2D = (X * \text{パース}) / (Z + \text{視点距離})$$

$$Y2D = (Y * \text{パース}) / (Z + \text{視点距離})$$

パースは通常 VIEW ポートの最大幅の2分の1をセットする。これによって、視野ピラミッドの幅が、VIEW ポートの端と一致する。計算は、ワールド座標で行われ、アプリケーションソフトは WINDOW と VIEW を使用して2次元クリッピングをおこなう。

<3次元ラインテーブル>

```
DW StartX1, StartY1, StartZ1
DW EndX1, EndY1, EndZ1  (, COLOR)
{
DW StartXn, StartYn, StartZn
DW EndXn, EndYn, EndZn  (, COLOR)
```

< 2次元ラインテーブル>

```

DW    StartX1, StartY1
DW    EndX1, EndY1, (, COLOR)
    {
DW    StartXn, StartYn
DW    EndXn, EndYn, (, COLOR)

```

39H	図形の回転
-----	-------

入 力 AH=38H

AL=コントロールデータ

DX=回転角度(0~359)

CX=ドットの数

DS:SI=回転前ラインテーブルアドレス(ファンクション 38 と同じ)

ES:DI=回転後ラインテーブルアドレス(ファンクション 38 と同じ)

INT 87H

出 力 なし

解 説 3次元(2次元)データを X, Y, Z 軸を中心に回転する。

<コントロールデータ>

bit 7:1=色コードあり

0=色コードなし

bit 6-2:00000

bit 1-0:回転軸

00 2次元データ

01 X軸回転

10 Y軸回転

11 Z軸回転

3AH	画面圧縮セーブ
-----	---------

入 力 AH=3AH

ES:DX=ファイル名のアドレス

INT 87H

出 力 DX: CX=圧縮データの長さ(32ビット)

AX=0:正常終了

1:ファイルの書き込み不能(ディスクフルなど)

3:ファイル名が無効

4:オープンされているファイルが多過ぎる

5:アクセス不能

9:ストップキーが押された

解 説 VIEW が区切られた GVRAM イメージを圧縮してファイルに書き出す。

3BH	画面再生ロード
-----	---------

入 力 AH=3BH

AL=描画開始座標の有無(1=有り/0=無し)

BX=描画開始 X 座標

CX=描画開始 Y 座標

ES:DX=ファイル名のアドレス

INT 87H

出 力 DX: CX=圧縮データの長さ(32 ビット)

AX=0: 正常終了

1: ファイルの書き込み不能(ディスクフルなど)

2: ファイルが見つからない

3: ファイル名が無効

4: オープンされているファイルが多過ぎる

5: アクセス不能

6: ファイルが正しくない

9: ストップキーが押された

解 説 画面圧縮セーブで圧縮したデータを GVRAM 上に再現する。描画はビットごとの論理 OR で行われるので、あらかじめ画面を消去しておく必要がある。また、VIEW ポートでのクリッピングは行わない。圧縮データは、画面モードが異なっても、その表示能力の範囲内で再現することができる。また、マルチプレーンとシングルプレーンの区別はない。

3CH	色の一括変更
-----	--------

入 力 AH=3CH

CX=変換する色の総数

ES: SI=変換前の色テーブル

ES: DI=変換後の色テーブル

INT 87H

出 力 なし

解 説 シングルプレーン 8・16 ビット/ピクセル時に画面中の色を一括して変更する。複数の色を一度に変換することができる。

<色テーブル>

○ 8 ビット/ピクセル時

ES: SI DB 変換前の色 1, 2, 3, 4, ……

ES: DI DB 変換後の色 1, 2, 3, 4, ……

○ 16 ビット/ピクセル時

ES: SI DW 変換前の色 1, 2, 3, 4, ……

ES: DI DW 変換後の色 1, 2, 3, 4, ……

3DH	ピクセルサイズ 16 → 8 変換
-----	-------------------

入 力 AH=3DH

CX = 8 ビット／ピクセル画面のフレームバッファの横幅

ES:DI = 8 ビット／ピクセル画面のフレームバッファの開始アドレス

INT 87H

出 力 なし

解 説 シングルプレーン 16 ビットピクセルのデータを 8 ビットピクセルのデータに変換する。

3EH	グラフィック画面拡大・縮小
-----	---------------

入 力 AH=3EH

ES:BX = 転送座標テーブルアドレス

INT 87H

出 力 なし

解 説 画面上の任意の範囲を拡大／縮小しながらコピーする。

<転送座標テーブル>

DW SX 1, SY 1, SX 2, SY 2

DW DX 1, DY 1, DX 2, DY 2

* (SX 1, SY 1) - (SX 2, SY 2) の領域を (DX 1, DY 1) - (DX 2, DY 2) に転送する。

範囲は 640×400 まで

6.8 アニメーション BIOS

アニメーション BIOS はアニメーションなどのアプリケーションプログラムで有用なツールを集めたものである。

●アニメーション BIOS 機能一覧

機能番号 (16 進)	機 能
00	直線の端点を指定する
01	直線の座標を得る
02	円の半径を指定する
03	円の座標を得る
04	スプライン曲線の元座標を指定する
05	スプライン曲線の座標を得る
06	イメージスキャナを初期化する
07	イメージスキャナの読取り開始指示
08	イメージスキャナからのデータ取得
09	イメージスキャナの読取り中止
0A	HSV 色指定を RGB 指定に変換する
0B	RGB 色指定を HSV 指定に変換する

●呼出し方

- ① レジスタ／メモリにパラメータ設定
- ② AH ← 機能番号
- ③ INT 88H
- ④ STI

●注意点

CLI 状態のまま戻ってくることがある。

00H	ライン座標設定
-----	---------

入 力 AH=00H

ES:BX=始点・終点座標を格納したテーブルへのポインタ
[ES:BX] 始点 X 座標(signed word)
[ES:BX+2] 始点 Y 座標(signed word)
[ES:BX+4] 終点 X 座標(signed word)
[ES:BX+6] 終点 Y 座標(signed word)
INT 88H

出 力 なし

解 説 直線の始点と終点を指定する。このファンクションで指定した直線の各点の座標を、ファンクション 1 で取得することができる。

01H	ライン座標取得
-----	---------

入 力 AH=01H
INT 88H

出 力 CY=1 ライン座標設定を行っていない。
CY=0 正常終了
AX=0 ラインの途中データ
=1 ラインの最後のデータ(終点)
BX=X 座標
CX=Y 座標

解 説 ファンクション0で設定した直線の各点の座標を、始点から終点に向かって1点ずつ取得する。このファンクションを1回呼ぶごとに点の座標を順に返す。

02H	円半径設定
-----	-------

入 力 AH=02H
DX=半径
INT 88H

出 力 CY=1 半径が0または負。
=0 正常終了

解 説 円の半径を指定する。この円の円周の座標をファンクション3で取得できる。

03H	円座標取得
-----	-------

入 力 AH=03H
INT 88H

出 力 CY=1 半径を設定していない。
=0 正常終了
AX=0 途中の点の座標
=1 最後の点の座標
BX=X 座標
CX=Y 座標

解 説 ファンクション2で指定された半径の1/8円の円周の座標を1点ずつ返す。円の中心は(0, 0)に固定されている。全円の座標は次のようにして得られる。
(SX, SY)中心点座標
(RX, RY)このファンクションで得た座標
とすると、

$(X0, Y0) = (SX + RX, SY + RY)$
 $(X1, Y1) = (SX + RY, SY - RX)$
 $(X1, Y2) = (SX - RX, SY - RY)$
 $(X3, Y3) = (SX - RY, SY + RX)$
 $RX \neq RY$ かつ $RX \neq 0$ のとき,
 $(X4, Y4) = (SX + RY, SY + RX)$
 $(X5, Y5) = (SX + RX, SY - RY)$
 $(X6, Y6) = (SX - RY, SY - RX)$
 $(X7, Y7) = (SX - RX, SY + RY)$
 の8または4点に点を打てばよい。

04H	スプライン元座標設定
-----	------------

入 力 AH=04H
 ES:BX=座標テーブルへのポインタ
 DX=座標の数(8以上)
 CX=各頂点(元座標)間で補間する数(1~1000)。
 INT 88H
 座標テーブルの形式
 ES:[BX] DW X0, Y0
 DW X1, Y1
 DW X2, Y2
)

出 力 CY=1 パラメータが異常
 =0 正常終了

解 説 3次Bスプライン曲線を計算するための頂点の座標テーブルを指定する。このテーブルはファンクション5で直接参照されるので、座標取得終了まで保持する必要がある。

05H	スプライン座標取得
-----	-----------

入 力 AH=05H
 INT 88H
出 力 CY=1 スプライン座標設定を行っていない。
 CY=0 正常終了
 AX=0 途中のデータ(まだ残りの座標がある)
 =1 最後のデータ(終点)
 BX=X座標
 CX=Y座標

解 説 ファンクション4で設定した頂点を元にした、3次Bスプライン曲線の座標データを1点ずつ取得する。誤差のため滑らかにならないことがある。

06H	イメージスキャナの初期化
-----	--------------

入 力 AH=06H
INT 88H

出 力 CY=1 スキャナの準備ができていない。
=0 正常終了

解 説 スキャナポートに接続されたスキャナを初期化する。スキャナは必ず接続されている必要がある。

07H	イメージスキャナ読取り開始
-----	---------------

入 力 AH=07H
CX=1 ラインのドット数をバイト数で指定(最大 240 バイト)
DX=読込みライン数
AL=読取りモード

Bit 7: ユーザ指定シーケンス (1=有り/0=無し)

Bit 6: 0

Bit 5, 4: ディザ指定 (2 階調のとき)

=00 スイッチによる

=01 強制的にディザ OFF

=10 ディザ指定

Bit 3, 2, 1, 0: 読取り階調

=0001 2 階調

=0010 4 階調

=0011 8 階調

=0100 16 階調

ES: BX=ユーザ指定シーケンスへのポインタ。

AL の Bit 7=1 のときのみ有効。

形式はスキャナへ出すエスケープシーケンスをそのまま格納し、NULL(00)で終了する。

INT 88H

出 力 CY=1 パラメータが異常
ファンクション 6 を行っていない
=0 正常終了

解 説 スキャナに読取りパラメータと読取り開始命令を送る。同時に各種エスケープシーケンスを指定して、多彩な設定をすることが可能となっている。

08H	イメージスキャナデータ取得
-----	---------------

入 力 AH=08H
ES: BX=データ格納領域ポインタ配列のアドレス
INT 88H

出 力 CY=1 読取りは終了している。
ファンクション 6, 7 を行っていない。
=0 正常終了
データ(1ライン分)が指定した領域に格納される。
AX=0 まだ残りのデータがある。
=1 最後のラインである。

解 説 スキャナから1ライン分のデータを読み込み、圧縮から復元して指定された領域に格納する。多階調の場合は、マルチプレーンタイプと同様な各階調ごとのプレーンとして格納される。この時格納されるプレーンの順番は、スキャナから送ってきた順である。

<データ格納領域ポインタ配列>
配列の形式は、1ラインのドット数/8(バイト)の領域への32ビットポインタを、
2階調のとき 1プレーン分
4階調のとき 2プレーン分
8階調のとき 3プレーン分
16階調のとき 4プレーン分
持ったもの。

<例: 16階調/幅320ドットのとき>
ES: BX → dw offset of plane 0
 dw segment of plane 0
 dw offset of plane 1
 dw segment of plane 1
 dw offset of plane 2
 dw segment of plane 2
 dw offset of plane 3
 dw segment of plane 3
plane0 db 40 dup(?)
plane1 db 40 dup(?)
plane2 db 40 dup(?)
plane3 db 40 dup(?)

09H	イメージスキャナ読取り中止
-----	---------------

入 力 AH=09H
INT 88H

出 力 なし

解 説 スキャナからの読取りを強制的に中止する。

0AH	HSV → RGB データ変換
-----	-----------------

入 力 AH=0AH
AL=RGB のタイプ
=0: 8 ビット/ピクセル
=1: 16 ビット/ピクセル
=2: フルビットモード (24 ビット)
BX=H (0<=H<=359)
CX=S (0<=S<=100)
DX=V (0<=V<=100)
INT 88H

出 力 CY=1 パラメータ異常
CY=0 正常終了
8 ビット/ピクセルのとき
BL=bit 7-5: G bit 4-2: R bit 1-0: B
16 ビット/ピクセルのとき
BX=bit 15-10: G bit 9-5: R bit 4-0: B
フルビットモードのとき
BX=B (0~63)
CX=R (0~63)
DX=G (0~63)

解 説 HSV 型で表された色データを RGB 型に変換する。
RGB 型にはグラフィックモードと一致する 8 ビット/ピクセル、16 ビット/ピクセルの他に、ハードウェア上の発色を反映したフルビットモードがある。

0BH	RGB → HSV データ変換
-----	-----------------

入 力 AH=0BH

AL=RGB のタイプ

=0: 8ビット／ピクセル

=1: 16ビット／ピクセル

=2: フルビットモード(24ビット)

8ビット／ピクセルのとき

BL= bit 7-5: G bit 4-2: R bit 1-0: B

16ビット／ピクセルのとき

BX= bit 15-10: G bit 9-5: R bit 4-0: B

フルビットモードのとき

BX=B(0~63)

CX=R(0~63)

DX=G(0~63)

INT 88H

出 力 CY=1 パラメータ異常

CY=0 正常終了

BX ← H(0 ≤ H ≤ 359)

CX ← S(0 ≤ S ≤ 100)

DX ← V(0 ≤ V ≤ 100)

解 説 RGB型で表された色データを HSV型に変換する。

6.9 プリンタ BIOS

プリンタ BIOS は、VA のプリンタ制御をサポートするためのものであり、11 個のファンクションがある。想定しているプリンタは、PC-PR 系の 24 ピンで、熱転写およびドットインパクト型のカラーまたは白黒のものである。

●プリンタ BIOS 機能一覧

機能番号 (16 進)	機 能
00	プリンタ BUSY チェック
01	2 バイトコード文字出力
02	画面ハードコピー
03	1 バイト出力
04	プリンタの種類の設定
05	設定されているプリンタの種類の取得
06	漢字・ANK の桁揃え指定
07	桁揃えモードの取得
08	タイムアウト時間の設定
09	ファンクションのリブレース
0A	作業領域先頭アドレスの取得

●呼出し方

- ① レジスタ／メモリにパラメータ設定
- ② AH ← 機能番号
- ③ INT 89H

●初期状態

- プリンタの種類 : SETUP 画面で設定されているプリンタ
- 漢字と ANK 文字の桁揃え : 桁揃え無し
- タイムアウトの時間 : 0 秒(タイムアウト無し)
- COPY キーでのコピー : 白黒反転モードのコピー

00H	プリンタ BUSY チェック
-----	----------------

入 力 AH=00H
INT 89H

出 力 AX=00 READY
=01 BUSY

解 説 プリンタの BUSY 信号をチェックする。

01H	2 バイトコード文字の出力
-----	---------------

入 力 AH=01H
DX=2 バイトコード
INT 89H

出 力 CF=0 正常終了
=1 エラー
AX=01 シフト JIS → JIS 変換のエラー
=02 STOP キーによる中断
=03 タイムアウトによる中断

解 説 2 バイトコード文字をプリンタに出力する。ANK 文字は1バイト、それ以外は2バイトのコードとなる。漢字のシフト JIS コードは JIS コードに変換されて出力される。TAB は必要な数のスペースに変換される。

02H	画面のハードコピー
-----	-----------

入 力 AH=02H
AL=コピーモード
bit 7: グラフィック画面1 (0=コピーする/1=しない)
bit 6: グラフィック画面0 (0=コピーする/1=しない)
bit 5: スプライト画面 (0=コピーする/1=しない)
bit 4: テキスト画面 (0=コピーする/1=しない)
bit 3: 0
bit 2: カラー/モノクロ濃淡 (0=カラー/1=モノクロ)
bit 1: 左右反転 (0=ノーマル/1=左右反転)
bit 0: 白黒反転 (0=ノーマル/1=白黒反転)
INT 89H

出 力 CF=0 正常終了
=1 STOP キーによる中断

解 説 合成画面のハードコピーをとる。カラーコピー時には64色で出力される。
※スプライトのコピーについて
①スプライト 31(キャラクタカーソル)は、コピーしない。
②スプライト 0-30(マウスカーソルを含む)は、表示されていればコピーする。
③このファンクションが実行されると、スプライト 0-30 は Sleep され、リターンすると Wake up する。

03H	1 バイト出力
-----	---------

入 力 AH=03H
DL=出力データ
INT 89H

出 力 CF=0 正常終了

=1 タイムアウトまたは STOP キーによる中断

解 説 1 バイトのデータをそのままプリンタに出力する。TAB の変換等はいっさい行われ
ない。

04H	プリンタ種類の設定
-----	-----------

入 力 AH=04H

AL=プリンタの種類

bit 1: 0=熱転写型/1=ドットインパクト型

bit 0: 0=カラー/1=モノクロ

INT 89H

出 力 なし

解 説 プリンタの種類を設定する。画面コピーファンクション使用時に必要となる。

05H	設定されているプリンタの種類の取得
-----	-------------------

入 力 AH=05H

INT 89H

出 力 AL=プリンタの種類

bit 1: 0=熱転写型/1=ドットインパクト型

bit 0: 0=カラー/1=モノクロ

解 説 設定されているプリンタの種類を取得する。

06H	漢字・ANK の桁揃え指定
-----	---------------

入 力 AH=06H

AL=00 桁揃えなし

=01 桁揃えあり

INT 89H

出 力 なし

解 説 ファンクション 01 で、漢字を印字するときに ANK 文字と桁を揃えるかどうかを設定する。桁揃えが指定されたときに漢字と漢字の間に 8 ドットスペースを空ける。

07H	桁揃えモードの取得
-----	-----------

入 力 AH=07H

INT 98H

出 力 AL=00 桁揃えなし
=01 桁揃えあり

解 説 桁揃えモードになっているかどうかを調べる。

08H	タイムアウト時間の設定
-----	-------------

入 力 AH=08H
DX=タイムアウト時間(秒) 0~7FFFH 指定
INT 89H

出 力 なし

解 説 プリンタのタイムアウト時間を設定する。DX=0 のときはタイムアウト無し(タイムアウト時間=無限大)として設定される。

09H	ファンクションのリプレイス
-----	---------------

入 力 AH=09H
ES:DX=ファンクションアドレステーブルへのポインタ

ES:DX+	各ルーチンのアドレス(2ワード)
+00	ファンクション 00
+04	ファンクション 01
+08	ファンクション 02
+0C	ファンクション 03
+10	カラー変換(パレット)
+14	カラー変換(RGB 8 bit)
+18	カラー変換(RGB 16 bit)
+1C	白黒濃淡変換(パレット)
+20	白黒濃淡変換(RGB 8 bit)
+24	白黒濃淡変換(RGB 16 bit)

INT 89H

出 力 なし

解 説 プリンタ BIOS の各ファンクションおよびカラー変換ルーチン(CRT カラー→プリンタリボンカラー)を取り換える。取り換える必要のないファンクションは、セグメント・オフセット共に 0 を指定する。

0AH	作業領域先頭アドレスの取得
-----	---------------

入 力 AH=0AH
INT 89H

出 力 ES:DX=プリンタ BIOS の作業領域アドレス

解 説 プリンタ BIOS の作業領域の先頭アドレスを返す。

6.10 コミュニケーション BIOS

コミュニケーション BIOS は RS-232 C を使った通信を処理する BIOS であり、22 個の機能が
ある。

●コミュニケーション BIOS 機能一覧

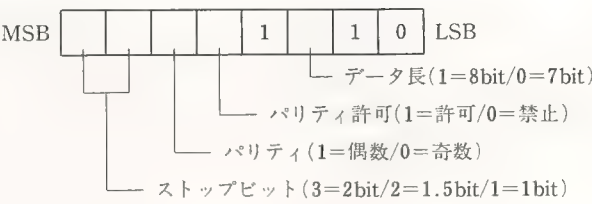
機能番号 (16 進)	機 能
00	通信環境の初期化
01	文字送信
02	文字の再送信
03	受信文字の取込み
04	受信文字のセンス
05	受信バッファを覗く
06	通信ステータスの読出し
07	BREAK 信号の送出
08	ボーレート設定
09	漢字コードの種類の設定
0A	通信方法の設定
0B	受信バッファの受信データ長を返す
0C	受信バッファのクリア
0D	制限時間付き文字送信
0E	コミュニケーション BIOS の解放
0F	XMODEM の開始
10	XMODEM の終了
11	XMODEM の 1 ブロック受信
12	XMODEM の 1 ブロック送信
13	受信割込みのトラップベクタ設定
14	ABORTKEY 1 の定義
15	ABORTKEY 2 の定義

●呼出し方

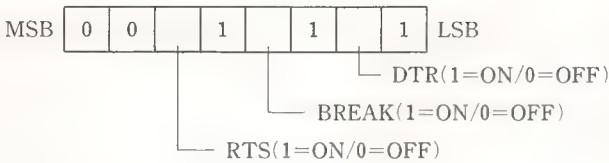
- ① レジスタ／メモリにパラメータ設定
- ② AH ← 機能番号
- ③ INT 8AH

00H	通信環境の初期化
-----	----------

入 力 AH=00H
DH=8251 モードバイト



DL=8251 コマンドバイト



INT 8AH

出 力 なし

解 説 8251 に対してソフトウェアリセット後、モード設定・コマンド設定を行い、受信バッファのポインタ・作業領域を初期化する。

※当 BIOS がサポートするモードは非同期(調歩同期)で、ビット長 8/7、クロックレート×16 である。また、入力パラメータのチェックは行わない。

<初期化状態>

ボーレート	: メモリスイッチに従う
漢字モード	: 送信—ビット長が 8 のときシフト JIS, 7 のとき NEC JIS 受信—漢字許可
XON/XOFF 制御	: 送受信とも行う
SI/SO 制御	: ビット長が 8 のときなし/ビット長が 7 のときあり
パリティエラー	: 有効

01H	文字送信
-----	------

入 力 AH=01H

DX=2 バイト文字コード

INT 8AH

出 力 CF=0:AL=0(正常終了)

=1:送信不可

AL=エラーステータス

bit 0: Xoff を受信しているため送信できない

bit 1: 8251 busy

bit 2: 途中で送信できなくなった(KI/KO の途中)

bit 3: 途中で送信できなくなった(漢字の 2 バイト目)

bit 7~4: 0

解 説 2 バイトコード文字を送信する。送信バイト数は一定ではない。データが ^S/^Q 以外の場合、Xoff を受信していればエラーを返す。8251 のステータスをチェックし、Busy の場合は約 140ms 監視を続け、140ms 以内に 8251 が ready となれば、データを 8251 に出力する。漢字データの場合は必要に応じて漢字コードの変換、KI/KO 付加を行う。送信できなかった場合は、そのデータを記憶しておき、再送信ファンクションに備える。

02H	文字の再送信
-----	--------

入 力 AH=02H
INT 8AH

出 力 CF=0: 正常終了
=1: 送信不可

解 説 ファンクション 01(1文字送信)できなかった文字列を再送信する。再送信すべき文字列が存在しない場合は、正常終了を返す。

03H	受信文字の読出し
-----	----------

入 力 AH=03H
INT 8AH

出 力 CF=0: 正常終了
DX=2バイト文字コード
AL=データステータス
bit 7: 通信エラー(1=あり/0=なし)
bit 6-2: 00000
bit 1-0: 受信した漢字コードの種類(DXの種類ではない)
00: シフト JIS
01: NEC JIS
10: 旧 JIS
11: 新 JIS
=1: データなし(バッファエンプティ)

解 説 漢字非使用モードでは、受信バッファの先頭バイトをそのまま返す。漢字使用モードでは、データが確定していれば2バイト文字コードに変換して返す。漢字データが未確定の場合はバッファエンプティを返す。
読出しポインタは更新される(destructive read)。

04H	受信文字のセンス
-----	----------

入 力 AH=04H
INT 8AH

出 力 CF=0: 正常終了
DX=2バイト文字コード
AL=データステータス(ファンクション 03と同じ)
=1: データなし(バッファエンプティ)

解 説 漢字非使用モードでは、受信バッファの先頭バイトをそのまま返す。漢字使用モードでは、データが確定していれば2バイト文字コードに変換して返す。漢字データが未確定の場合はバッファエンプティを返す。
読出しポインタは更新されない(non-destructive read)。

05H	受信バッファを覗く
-----	-----------

入 力 AH=05H
DH=受信バッファ先頭からのオフセット
INT 8AH

出力 CF=0: 正常終了
DL=受信バッファデータ
DH=変更なし
=1: データなし(バッファエンプティ)
DX=0

解 説 受信バッファ中のデータ(バイト)をそのまま返す。バッファ中のデータ数より大きいオフセットが与えられた場合にはバッファエンプティを返す。

06H	通信ステータスの読正し
-----	-------------

入 力 AH=06H
INT 8AH

出 力 DL=8251 ステータス
bit 7: DSR(DR)
bit 6: break detect
bit 5: 0(Flaming error)
bit 4: 0(Overrun error)
bit 3: 0(Parity error)
bit 2: Tx empty
bit 1: Rx ready
bit 0: Tx ready
DH=XON/XOFF 状態および制御線状態
bit 7: CI
bit 6: CTS(CS)
bit 5: DCD(CD)
bit 4~1: 0000
bit 0: Xoff receive

解 説 各種通信状態を調べる。

07H	BREAK 信号の送出
-----	-------------

入 力 AH=07H
DH=送出時間(0.5 秒単位, 1-7)
INT 8AH

出 力 なし

解 説 BREAK 信号を最大 3.5 秒間送出する。送出完了まで戻らない。
DH=0 の場合は直ちに BREAK OFF して戻る。

08H	ボーレートの設定
-----	----------

入力 AH=08H
DH=ボーレートファクタ(0-9)
INT 8AH

出力 CF=0: 正常終了
=1: ボーレートファクタが間違っている

解 説 ボーレートファクタから分周値を求め、クロック発生器(TCU カウンタ#1)と作業領域に設定する。ボーレートファクタ=0 の場合は、作業領域に設定されている分周値をそのままクロック発生器に設定するため、自由なボーレートの設定も可能である。

ファクタ	ボーレート	分周値
0	free	作業領域設定値
1	75	0D00H
2	150	0680H
3	300	0340H
4	600	01A0H
5	1200	00D0H
6	2400	0068H
7	4800	0034H
8	9600	001AH
9	19200	000DH

09H	漢字コード種類の設定
-----	------------

入 力 AH=09H
DH=漢字コードの種類

bit 7: 1=受信時の漢字非使用/0=使用

bit 6-2: 00000

bit 1-0: 送信漢字コードの種類
00=シフト JIS
01=NEC JIS
10=旧 JIS
11=新 JIS

INT 8AH

出 力 CF=0: 正常終了
 =1: パラメータ異常

解 説 送信時の漢字コード種類の設定, 受信時の漢字使用の有無を設定する。

0AH	通信制御の設定
-----	----------------

入 力 AH=0AH
 DH=セットアップ情報
 bit 7: パリティエラー制御(1=エラー無視/0=有効)
 bit 6-4: 000
 bit 3: 受信時 SI/SO 制御(1=する/0=しない)
 bit 2: 送信時 SI/SO 制御(1=する/0=しない)
 bit 1: 受信時 XON/XOFF 制御(1=する/0=しない)
 bit 0: 送信時 XON/XOFF 制御(1=する/0=しない)

INT 8AH

出 力 なし

解 説 通信制御モードを設定する。

0BH	受信バッファデータ数取得
-----	---------------------

入 力 AH=0BH
 INT 8AH

出 力 DX=受信バッファ内のデータ数(バイト)

解 説 受信バッファ内のデータ数(バイト)を得る。

0CH	受信バッファのクリア
-----	-------------------

入 力 AH=0CH
 INT 8AH

出 力 なし

解 説 受信バッファをクリアする。

0DH	時間制限付き 1 バイト受信
-----	-----------------------

入 力 AH=0DH
 DX=制限時間(100 ms 単位)
 INT 8AH

出力 CF=0: 受信成功
 DL=受信データ
 DH=0
 =1: 受信失敗
 AX=FFFFH: タイムアウト(DX=0)
 =キーコード: アボートキー押下(DX=残り時間)

解説 受信バッファにデータがあれば先頭の1バイトを返す。バッファが空のときには制限時間まで受信を待つ。途中でアボートキーが押されたときは、直ちに戻る。

0EH	コミュニケーション BIOS の解放
-----	--------------------

入力 AH=0EH
 INT 8AH

出力 なし

解説 コミュニケーション BIOS をスリープモードに移行させる。ファンクション 00 H でアクティブになる。

0FH	XMODEM の開始
-----	------------

入力 AH=0FH
 DH=00H: 受信(Open for receive)
 =FFH: 送信(Open for transmit)
 INT 8AH

出力 CF=0: DL=0 エラーなし
 =1: DL=1 XMODEM は使用できない(データ長=7ビット)
 DL=2 タイムアウト
 DL=3 アボートキー押下

解説 XMODEM 使用開始を宣言し、BIOS を XMODEM 状態にする。送信時は 60 秒のタイムアウトの範囲内で相手からの NAK を待つ。途中でアボートキーが押されたら直ちに XMODEM から抜け出す。また、XMODEM 使用時は、XON/XOFF 制御・SI/SO 制御・パリティエラー無視は行わない。

10H	XMODEM の終了
-----	------------

入力 AH=10H
 DH=00H: 正常終了
 =18H: アボート終了(CAN を送出する)
 =FFH: アボート終了(何も送出しない)
 INT 8AH

出力 CF=0: エラーなし
 =1: アボート／受信側からの ACK がない。

解説 XMODEM 使用終了を宣言し、BIOS を XMODEM 状態から抜け出させる。正常終了指定の場合、送信時は EOT を送信後 ACK を待つ、受信時は ACK を送信する。

11H	XMODEM1 ブロック受信
-----	----------------

入力 AH=11H
 AL=ACK(06H) or NAK(15H)
 ES: DX=データ転送アドレス
 INT 8AH

出力 CF=0: エラーなし
 DL=00: 連続ブロック (DH=ブロック番号)
 =04: ブロック終了
 =1: エラー
 DL=01: バイト間タイムアウト
 DL=02: ブロックタイムアウト
 DL=03: BCC エラー／ブロック#サムエラー／バイトエラー
 DL=05: CAN 受信
 DL=06: アボートキー押下

解説 ACK/NAK を送信し、ブロックタイムアウトの範囲で相手からのデータを待つ。ブロックの誤りをチェックし、誤りが無い場合、指定されたアドレスにデータを転送する。ブロック#の正当性チェックはアプリケーションソフトが行う。また、アボートキーが押された場合は、直ちに処理を中断する。

12H	XMODEM1 ブロック送信
-----	----------------

入力 AH=12H
 AL=送信ブロック番号
 ES: DX=データ転送アドレス
 INT 8AH

出力 CF=0: エラーなし (ACK 受信) DL=00
 =1: エラー
 DL=01: NAK 受信
 DL=02: タイムアウト
 DL=03: 8251 ビジー
 DL=05: CAN 受信
 DL=06: アボートキー押下

解説 ヘッド・BCCを生成し、データと共に送信する。送信ブロックデータ長は128バイト固定である。

- ①受信バッファにCANがあるかどうか調べる
 - ②ヘッド，BCCを生成し，送信する
 - ③ブロック送信後直ちに受信バッファをクリアする
 - ④ブロックタイムアウト内でACK／NAKを待つ
 - ⑤結果を返す
- また，アボートキーが押された場合は，直ちに処理を中断する。

13H	受信割込みのトラップアドレス設定
-----	-------------------------

入力 AH=13H
 ES: DX=トラップルーチンアドレス (ES=0はトラップOFF)
 INT 8AH

出力 なし

解説 受信割込み発生時のトラップルーチンを登録・消除する。
 トラップ時のレジスタは次のようになっており，破壊してはならない。
 AL=受信データ(バイト)
 DX=0020H
 DS=ES=0040H
 Flag=CLI状態

14H	アボートキー#1の定義
-----	--------------------

入力 AH=14H
 DX=アボートキー#1のキーコード
 INT 8AH

出力 DX=以前のアボートキー#1

解説 通信の中断を指示するアボートキーの#1を定義する。

15H	アボートキー#2の定義
-----	--------------------

入力 AH=15H
 DX=アボートキー#2のキーコード (FFFFHで定義削除)
 INT 8AH

出力 DX=以前のアボートキー#2

解説 通信の中断を指示するアボートキーの#2を定義する。

作業領域 00CBH : 0000H～

コミュニケーション BIOS の作業領域のセグメントは 00CBH である。

OFFSET (16 進)	
00-03	受信割込み時のトラップアドレス(ファンクション 13 H 参照)
04-05	アボートキー#1 (デフォルトはC 2B03H)
06-07	アボートキー#2 (デフォルトは STOP 6003H)
08-13	内部使用
14-15	クロック発生器に設定した分周値(ファンクション 08H 参照)
16	当 BIOS アクティブフラグ(00H=sleep/FFH=active)
17	ポーレートファクタ(ファンクション 08H 参照)
18	漢字コードの種類(ファンクション 09H 参照)
19	セットアップ情報(ファンクション 0AH 参照) ファンクション 0AH での定義の外に次の 2 ビットがある。 bit 5: 1=XMODEM/0=normal bit 4: 1=XMODEM 送信/0=XMODEM 受信
1A	フローズステータス(逐次更新される) bit 0: 送信 XON/XOFF 状態(1=XOFF/0=XON) bit 1: 受信 XON/XOFF 状態(1=XOFF/0=XON) bit 2: 送信 SI/SO 状態(1=SO/0=SI) bit 3: 受信 SI/SO 状態(1=SO/0=SI) bit 4: 送信 KI/KO 状態(1=KI/0=KO) bit 5: 受信 KI/KO 状態(1=KI/0=KO) bit 7~6: 00
1B	XMODEM でブロック受信中のバイト間タイムアウトの時間 (100ms 単位で 100ms~25.5 秒 デフォルト=1 秒)
1C	XMODEM でブロック間のタイムアウトの時間 (100ms 単位で 100ms~25.5 秒 デフォルト=10 秒)
この他に当 BIOS の作業領域ではないが、以下の変数を参照/設定している。	
0040 : 005E	8251 に設定したモードバイト 原則としてこの変数はファンクション 00 により BIOS が更新する
0040 : 005F	8251 に設定したコマンドバイト アプリケーションソフトが 8251 を制御する際には、8251 に OUT すると共にこのバイトを更新する。
0040 : 0CBD-0CBE	タイマー割込みが 1/120 秒ごとに減算し続けるカウンタ

6.11 サウンド BIOS

サウンド BIOS は、YM2203(OPN)を使用して演奏を行うものである。実行形態としては2種類あり、1つはコマンド識別コードにより各処理へ分岐し、処理終了後リターンする形態、もう一つは BASIC の PLAY 文と同様コマンド列を PLAY バッファに格納するのみでリターンし、インターバルタイマにより割込みで演奏を行う形態である。

2つのインターバルタイマのうち、タイマ A は通常の演奏用(テンポ)に、タイマ B は、LFO 効果の制御用(暫定的 4ms 固定)に使用している。

●サウンド BIOS 機能一覧

機能番号 (16 進)	機 能
00	サウンド BIOS 初期化
01	演奏開始
02	演奏中止
03	レジスタ読出し
04	レジスタ書込み
05	音長割合(G/S 値)の設定
06	音の発生
07	デフォルト音長の設定
08	テンポの設定
09	音色の設定
0A	音色パラメータの個別読出し
0B	音色パラメータの個別設定
0C	演奏の一時中断
0D	演奏再開
0E	LFO 開始
0F	LFO 中止
10	KEY 状態維持
11	PLAY 割込み条件の設定
12	音量設定
13	BGM モード指定
14	MIDI 出力
15	エンベロープ設定
16	演奏モードの設定
17	1 機能単位 of データ格納
18	演奏状態の取得
19	音色データの一括読出し
1A	キュー残量の取得
1B	ブロックデータ演奏
1C	作業領域アドレスの取得

●呼出し方

- ① パラメータ設定
- ② AH ← 機能番号
- ③ INT 8BH

00H	サウンド BIOS 初期化
-----	---------------

入 力 AH=00 H

AL=00：同期演奏モード

01：独立演奏モード

DX=1 チャネル当りの PLAY キューサイズ(10H~2A00H)

DS=PLAY キューセグメントアドレス

INT 8BH

出 力 なし

解 説 OPN およびサウンド BIOS の初期設定を行う。サウンド BIOS を使用する時は、必ず最初にこのファンクションを実行しなければならない。

ES で指定されたセグメントから DX で指定された大きさのキューが 6 チャネル分 PLAY キューとして使用される。サウンド機能の使用終了まで、PLAY キューエリアは確保しておく必要がある。AL で指定したモードは、ファンクション 01 H・17H・1BH で有効

<初期設定値>

テンポ : 120(120 * 48 Steps/min)

4 分音符の長さ : 48 Steps

音長割合(G/S 値) : 8/8

CH 0-2 : 音色=ハープシコード 音量=64

CH 3-5 : 音色=不定 音量=8 エンベロープ周期=255

01H	演奏開始
-----	------

入 力 AH=01 H

AL=00：同期データ付加

01：同期データ付加しない

ES:BP=パラメータリストアドレス

INT 8BH

出 力 なし

解 説 音楽データ列を演奏する。28 バイトのパラメータリストを参照して、各チャネルのデータを PLAY キューに転送し、演奏を開始する。BGM モードの時はそのままリターンし、BGM モードでない時は演奏が終るまで待つ(ファンクション 13H 参照)。PLAY データ列はキューの大きさより小さくならない。データ列が正しくない場合の動作は保証しない。独立演奏モードでは同期データを付加しても無視される。

<パラメータリスト>

DW データ列セグメントアドレス

DW 0

DW CH1 データ列オフセット, CH1 データ長

DW CH2 データ列オフセット, CH2 データ長
 DW CH3 データ列オフセット, CH3 データ長
 DW CH4 データ列オフセット, CH4 データ長
 DW CH5 データ列オフセット, CH5 データ長
 DW CH6 データ列オフセット, CH6 データ長
 データ列の形式は当 BIOS の最後を参照。

02H	演奏中止
-----	------

入 力 AH=02H
 AL=00: 作業領域の初期化はしない
 =01: 作業領域の初期化もする
 INT 8BH

出 力 なし

解 説 現在の演奏を中止し, PLAY キューをクリアする。また, AL=1 の時は, BIOS の作業領域も初期化する。インターバルタイマの割込みも停止する。

03H	レジスタ読出し
-----	---------

入 力 AH=03H
 AL=レジスタ番号
 INT 8BH

出 力 DL=レジスタ内容

解 説 OPN のレジスタに設定されているデータを読み出す。レジスタ 21H 以上は OPN から読み出せないため各ファンクションにより出力された値を保存しておき, このファンクションではその値を返す。したがって, 直接ハードウェアに書き込んだデータについては, 保証しない。

04H	レジスタ書込み
-----	---------

入 力 AH=04H
 AL=レジスタ番号
 DL=レジスタ内容
 INT 8BH

出 力 なし

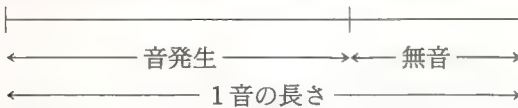
解 説 指定された OPN のレジスタデータを出力し, 同時にワークエリアに保存する。本ファンクションによってタイマ等サウンド BIOS の基本動作に関するレジスタを書き換えた場合, その後の BIOS の動作については保証されない。

05H	音長割合(G/S 値)の設定
-----	----------------

入 力 AH=05H
CH=チャンネル番号(0-5)
DL=音長割合(0-7:1/8~8/8 に対応)
INT 8BH

出 力 なし

解 説 1 音の長さの中で、実際に音を発生している時間の割合を設定する。



割合は 1/8 から 8/8 までを指定できる。1 音が短いときには、必ずしも正確な割合で発生するわけではない。V3 BASIC の PLAY 文 Q コマンドに相当する。

06H	音の発生
-----	------

入 力 AH=06H
CH=チャンネル番号(0-5)
DH=音程
70 H: 休符
MSB: 0=G/S 値の長さで KEY OFF する
1=G/S 値を無視し、KEY OFF しない(タイ)
bit 6-0: 音程指定(00H-60H)
DL=音長(01-FFH, Step Time 単位) 00=デフォルト音長
INT 8BH

出 力 なし

解 説 指定した音を発生する。現在の音が終わっていても新しく発生する。
DH の bit 6-0 で指定する値と実際の音程との関係は V3 BASIC の PLAY 文 N コマンドと同じ。

07H	デフォルト音長の設定
-----	------------

入 力 AH=07H
CH=チャンネル番号(0-5)
DL=音長(01-FFH, Step Time 単位)
INT 8BH

出 力 なし

解 説 指定チャンネルのデフォルト音長を設定する。

08H	テンポの設定
-----	--------

入 力 AH=08H
DL=テンポ(01-FFH)
INT 8BH

出 力 なし

解 説 演奏のテンポを設定する。実際にはインターバルタイマの期間を再設定する。テンポの単位は1分間に演奏できる4分音符の個数である。

09H	音色の設定
-----	-------

入 力 AH=09H
CH=チャンネル番号(0-5)
DL=音色パラメータブロック形式
00:ワード
01:バイト
80-BDH:ROMデータ(@0-@61)
ES:BP=音色パラメータブロックアドレス(DL=00/01時のみ)
INT 8BH

出 力 なし

解 説 指定チャンネルの音色を変更する。当BIOSが持つデータ(ROMデータ)を指定した場合にはES:BPは必要ない。このファンクションを実行するとLFO効果は、ONになり、LFOパラメータの設定が行われる。指定チャンネルがSSG音源のときは、LFOパラメータ(Amd, Ams以外)以外は、意味を持たない。

0AH	音色パラメータの個別読出し
-----	---------------

入 力 AH=0AH
AL=パラメータ番号(00-31H)
CH=チャンネル番号(0-5)
INT 8BH

出 力 DX=設定されている値

解 説 指定チャンネルの指定パラメータをワークエリアより読み出す。SSG音源においては、LFOパラメータ(Amd, Ams以外)は無意味である。パラメータがバイトの場合、DHはクリアされる。

0B	音色パラメータの個別設定
----	--------------

入 力 AH=0BH
AL=パラメータ番号(00-31H)
CH=チャンネル番号(0-5)
DX=設定値
INT 8BH

出 力 なし

解 説 指定チャンネルの指定パラメータを設定する。

0CH	演奏の一時中断
-----	---------

入 力 AH=0CH
INT 8BH

出 力 なし

解 説 BGM モードで演奏中に一時中断する。全チャンネルの KEY ON/OFF 情報を保存後、全チャンネルを KEY OFF する。インターバルタイマの割込みも禁止する。演奏を再開するには、ファンクション 01H・17H・0DH を実行する。
ワークエリアは、保証されていない。

0DH	演奏再開
-----	------

入 力 AH=0DH
INT 8BH

出 力 なし

解説 ファンクション 0CH で中断した演奏を再開する。ファンクション 0CH が実行されていなければ、何もしない。

0EH	LFO 開始
-----	--------

入 力 AH=0EH
CH=チャンネル番号(0-5)
INT 8BH

出 力 なし

解 説 指定チャンネルの LFO 機能を ON にする。LFO パラメータが無効であれば、何もしない。

0FH	LFO 中止
-----	--------

入 力 AH=0FH
CH=チャンネル番号(0-5)
INT 8BH

出 力 なし

解 説 指定チャンネルの LFO 機能を OFF にする。

10H	KEY 状態維持
-----	----------

入 力 AH=10H
CH=チャンネル番号(0-5)
DL=音長(01-FFH, Step time 単位)
00 のときはデフォルト長
INT 8BH

出 力 なし

解説 指定チャンネルの KEY ON/OFF 状態を指定した時間だけ維持する。

11H	PLAY 割込み条件の設定
-----	---------------

入 力 AH=11H
CH=チャンネル番号(0-5)
DX=割込みモード/データ長
MSB: 0=PLAY 割込み禁止/1=PLAY 割込み許可
bit 14-0: 割り込みを発生させるデータ長
ES:BP=割込みサービスルーチンエントリ
INT 8BH

出 力 なし

解 説 指定チャンネルの PLAY バッファ中のデータが少なくなった時の割込み条件を設定する。割込みルーチンは、FAR CALL で呼ばれ、割込み禁止の状態で実行される。割込みルーチンの実行時間が長いとテンポが狂う等の障害がおこる場合がある。割込みルーチンに渡されるレジスタは、以下の通りである。またレジスタは保証する必要はない。

CH=チャンネル番号(0-5)
DX=残りデータ数

12H	音量設定
-----	------

入 力 AH=12H
 CH=チャンネル番号(0-5)
 DL=音量(00-7FH)
 INT 8BH

出 力 なし

解 説 演奏の音量を変更する。FM 音源では指定チャンネルのアルゴリズムを参照し、出力オペレータの設定を変える。SSG 音源の時には、0-0FH のみ有効である。またエンベロープの指定はファンクション 15H で行い本ファンクションでは対応しない。

13H	BGM モード指定
-----	-----------

入 力 AH=13 H
 AL=0：非 BGM モード(演奏終了まで待つ)
 =1：BGM モード
 INT 8BH

出 力 なし

解 説 ファンクション 01H(PLAY)時の演奏モードを指定する。

14H	MIDI 出力
-----	---------

入 力 AH=14H
 CL=チャンネル番号(0-5)
 DL=出力データ
 INT 8BH

出 力 なし

解 説 MIDI I/O にデータを出力するためのファンクションであるが、拡張命令であり、現在は何もしない。

15H	エンベロープ設定
-----	----------

入 力 AH=15H
 CL=チャンネル番号(3-5)
 AL=パラメータ種類
 00： エンベロープ形状
 01： エンベロープ周期

DX=エンベロープ形状または周期

形状： 0-0FH(DL レジスタのみで可)

周期： 0-FFFFH

INT 8BH

出 力 なし

解 説 SSG 音源のエンベロープに関するデータを設定する。FM 音源を指定した場合は、何もしない。エンベロープ形状は次の KEY ON 時に有効となり、エンベロープ周期はそのまま OPN に設定される。エンベロープ形状を指定した時は、以後ファンクション 12H(音量設定)を行うまでエンベロープが有効となる。

16H	演奏モードの設定
-----	-----------------

入 力 AH=16H

DL=01: MIDI

02: FM+SSG

03: 効果音モード

04: CSM モード

INT 8BH

出 力 なし

解 説 演奏の出力状態を設定する。デフォルトは、FM+SSG モードである。

17H	1 機能単位のデータ格納
-----	---------------------

入 力 AH=17H

CH=チャンネル番号(0-5)

DL=データ長(01-FFH)

00 のときは演奏開始を示す

ES: BP=格納データアドレス

INT 8BH

出 力 AL=00H: 正常終了

=01H: PLAY キューが FULL で入らなかった

解 説 指定チャンネルの PLAY キューにデータを入れる。入力されたデータは演奏中であれば、そのまま実行される。演奏中でなければキューに格納されるだけで実行されないため、全チャンネルのデータを入力後、DL=0 として演奏開始を指示する必要がある。また、同期モードで実行するには、各チャンネルのデータ群に最後に同期データ(FFH)を入れる。同期データはデータの入力されていないチャンネルにも入れなければならない。

18H	演奏状態の取得
-----	---------

入 力 AH=18H
INT 8BH

出 力 AL=00: 演奏は終了している
01: PLAY キューは空になったが, 演奏中
02: PLAY キューにデータがあり, 演奏中
03: PLAY キューにデータがあるが, 演奏停止中

解 説 現在の演奏状態を知るためのファンクションである。BGM モードでの演奏終了や、LFO 開始のタイミングなどに利用できる。

19H	音色データの一括読み出し
-----	--------------

入 力 AH=19H
CH=音色指定
0-5: チャンネルに設定されている音色
80-BDH: ROM データ@0~@61 の音色
DL=パラメータ形式
00: ワード
01: バイト
ES: BP=転送先アドレス
INT 8BH

出 力 AL=00: 演奏は終了している
01: PLAY キューは空になったが, 演奏中
02: PLAY キューにデータがあり, 演奏中
03: PLAY キューにデータがあるが, 演奏停止中

解 説 ファンクション 09H と逆の動作をするものであり、音色データをメモリに読み出す。データ量は、バイト形式で 51 バイト、ワード形式で 100 バイトである。

1AH	キュー残量の取得
-----	----------

入 力 AH=1AH
CH=チャンネル番号(0-5)
INT 8BH

出 力 DX=指定チャンネルのキューの残量(バイト)

解 説 指定チャンネルの残り PLAY キューサイズを得る

1BH	ブロックデータ演奏
-----	-----------

入 力 AH=1BH

AL=繰り返し回数(0=永久ループ)

ES:BP=パラメータリストアドレス(ファンクション 01 と同じ)

INT 8BH

出 力 なし

解 説 パラメータリストで示されたデータ領域の PLAY データを演奏する。ファンクション 01 と異なり、データをキューにコピーせず、与えられたデータを直接演奏する。したがって、データ長はキューの大きさに関係なくなるが、演奏中はパラメータリストとデータ領域を保証しておかなければならない(変えるとそのデータで演奏される)。繰り返し回数を 0 にすると永久に繰り返すが、これを止めるにはファンクション 02H・0CH を実行する必要がある。本ファンクション実行中は PLAY キューがパラメータリストで指定されたデータ列に設定されるので、キューを変更するファンクションを使用すると、データも変更されてしまう。同期モードで演奏するときは、同期データ(FFH)をあらかじめデータに入れておく必要がある。

1CH	作業領域アドレスの取得
-----	-------------

入 力 AH=1CH

INT 8BH

出 力 ES:DX=作業領域の先頭アドレス

解 説 当 BIOS が使用している作業領域の先頭アドレスを示す。

音楽データ例

ファンクション 01H・1BH でパラメータリストが指すデータ列、ファンクション 17H でキューに格納するデータの形式を示す。各機能は 2～6 バイトで構成されており、先頭の 1 バイトで機能が区別される。各機能には対応するファンクションがあり、機能名の後の [] でファンクション番号を示す。パラメータの内容等についてはそのファンクションを参照すること。

第1バイト (16進)	機 能
00-60	音程データ
70	休符
80-E0	音程データ 第2バイト：音長(00H, 01-FFH)
F1	レジスタ書込み [04H] 第2バイト：OPN レジスタ番号 第3バイト：OPN 設定値
F2	音長割合(G/S 値)の設定 [05H] 第2バイト：G/S 値(0-7H)
F3	デフォルト音長の設定 [07H] 第2バイト：音長(0-FFH)
F4	テンポの設定 [08H] 第2バイト：テンポ(01-FFH)
F5	音色の設定 [09H] 第2バイト：パラメータリスト形式(00H/01H/80H-BDH) 第2バイトが 00H/01H のとき、 第3, 4バイト パラメータリスト開始オフセット(L, H) 第5, 6バイト パラメータリスト開始セグメント(L, H)
F6	音色パラメータの個別設定 [0BH] 第2バイト：パラメータ番号(0-31H) 3, 4バイト：設定値(L, H)
F7	LFO 開始/中止 [0EH/0FH] 第2バイト：01H=開始 00H=中止
F8	KEY 状態維持 [10H] 第2バイト：音長
F9	音量設定 [12H] 第2バイト：音量設定値(00-7FH)
FA	MIDI 出力 [14H] 第2バイト：出力データ
FB	エンベロープ設定 [15H] 第2バイト：00H=形状指定/01H=エンベロープ指定 第3, 4バイト：形状(0-FFH)第4バイトはない 周期(0-FFFFH)
FC	演奏モードの設定 [16H] 第2バイト：演奏モード(1-4)

6.12 マウス BIOS

マウス BIOS はスプライトを用いたマウスを制御する BIOS であり，以下の 14 個の機能を持つ。

●マウス BIOS 機能一覧

機能番号 (16 進)	機 能
00	マウスの初期化
01	マウスカーソルの表示開始
02	マウスカーソルの表示停止
03	マウス情報の取得
04	マウスカーソルの位置指定
05	左ボタンの押下情報の取得
06	左ボタンの解放情報の取得
07	右ボタンの押下情報の取得
08	右ボタンの解放情報の取得
09	マウスカーソルの形状指定
0A	マウス移動量を得る
0B	マウストラップの設定
0C	マウス移動比の設定
0D	マウスの水平移動範囲の設定
0E	マウスの垂直移動範囲の設定
0F	マウスをスリープ状態にする
10	マウスをアクティブ状態にする
11	マウス状態の取得
12	マウスの読出し周期の設定

●呼び出し方

- ①レジスタ／メモリにパラメータ設定
- ②AH ← 機能番号
- ③INT 33H

VA のマウスは他機種と比べて次のような特徴がある。

- ① 16 色／モノクロのマウスカーソルが選べる
- ②マウスカーソルの大きさがかなり自由である
 - 縦-4 ドットごとに 248 ドットまで
 - 横-16 色のとき 8 ドット，モノクロのとき 32 ドットごとに 256 ドット未満で，かつデータ量 512 バイトまで

00H	マウスの初期化
-----	---------

入 力 AX=00H
INT 33H

出 力 AX=FFFFH：正常終了
=0000H：マウスが使用できない

解 説 マウスの使用環境を初期化する。このファンクションはスプライト BIOS の初期化ファンクションからも呼ばれる。

- 1) マウスカーソルの消去
- 2) マウスカーソルの位置(0, 0)
- 3) マウスカーソルの中心は左上
- 4) ボタンの押離回数のクリア
- 5) スプライト BIOS に合わせた座標系の決定(640×400 or 640×200)
- 6) マウス移動範囲の初期化
- 7) マウスカーソル移動比(mouse/pixel)の初期化
(400 ライン時 8, 200 ライン時 16)
- 8) トラップの禁止
- 9) マウスカーソル形の初期化(color, 16×16, Arrow)
- 10) マウスリードタイミングの初期化(16 ms)

01H	マウスカーソルの表示開始
-----	--------------

入 力 AX=01H
INT 33H

出 力 なし

解 説 マウスカーソルの表示を開始する。

02H	マウスカーソルの表示停止
-----	--------------

入 力 AX=02H
INT 33H

出 力 なし

解 説 マウスカーソルの表示を停止する。

03H	マウス情報の取得
-----	----------

入 力 AX=03H
INT 33H

出力 AX=左ボタンの状態(FFFFH=押/0000H=離)
 BX=右ボタンの状態(FFFFH=押/0000H=離)
 CX=水平位置(0-639)
 DX=垂直位置(0-199 or 0-399)

解説 マウスカーソル位置とボタンの状態を調べる。

04H	マウスカーソルの位置指定
-----	--------------

入力 AX=04H
 CX=水平位置(0-639)
 DX=垂直位置(0-199 or 0-399)
 INT 33H

出力 なし

解説 マウスカーソル位置を指定する。

05H	左ボタンの押下情報の取得
-----	--------------

入力 AX=05H
 INT 33H

出力 AX=左ボタンの現在の状態(FFFFH=押/0000H=離)
 BX=前回このファンクションが呼ばれてから左ボタンが押された回数
 CX=最後に左ボタンが押されたときの水平位置(0-639)
 DX=最後に左ボタンが押されたときの垂直位置(0-199 or 0-399)

解説 左ボタンの押下に関する情報を得る。

06H	左ボタンの解放情報の取得
-----	--------------

入力 AX=06H
 INT 33H

出力 AX=左ボタンの現在の状態(FFFFH=押/0000H=離)
 BX=前回このファンクションが呼ばれてから左ボタンが離された回数
 CX=最後に左ボタンが離されたときの水平位置(0-639)
 DX=最後に左ボタンが離されたときの垂直位置(0-199 or 0-399)

解説 左ボタンの解放に関する情報を得る。

07H	右ボタンの押下情報の取得
-----	--------------

入 力 AX=07H
INT 33H

出 力 AX=右ボタンの現在の状態(FFFFH=押/0000H=離)
BX=前回このファンクションが呼ばれてから右ボタンが押された回数
CX=最後に右ボタンが押されたときの水平位置(0-639)
DX=最後に右ボタンが押されたときの垂直位置(0-199 or 0-399)

解 説 右ボタンの押下に関する情報を得る。

08H	右ボタンの解放情報の取得
-----	--------------

入 力 AX=08H
INT 33H

出 力 AX=右ボタンの現在の状態(FFFFH=押/0000H=離)
BX=前回このファンクションが呼ばれてから右ボタンが離された回数
CX=最後に右ボタンが離されたときの水平位置(0-639)
DX=最後に右ボタンが離されたときの垂直位置(0-199 or 0-399)

解 説 右ボタンの解放に関する情報を得る。

09H	マウスカーソルの形状指定
-----	--------------

入 力 AX=09H
BX=マウスカーソルセンタの X 座標
CX=マウスカーソルセンタの Y 座標
ES:DX=マウスカーソルタイルへのポインタ
INT 33H

出 力 AX=FFFFH:正常終了
=0000H:マウスカーソルが大き過ぎる

解 説 マウスカーソルの形状とカーソルセンタを設定する。マウスカーソルの大きさには次の制限がある。

高さ: 4 ピクセルごとで 256 ドット未満
横幅: 16 色のとき 8 ピクセルドットごとで
1 色のとき 32 ピクセルドットごとで
256 ドット未満

データ量: 512 バイトまで

データ量: =16 色のとき 横幅×高さ/2 (バイト)
1 色のとき 横幅×高さ/8 (バイト)

<マウスカーソルスタイル> [V3 BASIC の MOUSE 2 文と同様]

dw 横幅(ピクセル)

dw 高さ(ピクセル)

dw 色モード(1または4)

dw カラーコード(0~16:色モード=1のとき)

dw マウスカーソルパターン...

{ }

マウスカーソルパターンの格納アドレスは、TVRAM の AFD00H-AFEFFH.

0AH	マウス移動量を得る
-----	-----------

入 力 AX=0AH

INT 33H

出 力 CX=前回このファンクションが呼ばれてからの水平移動量

DX=前回このファンクションが呼ばれてからの垂直移動量

解 説 前回このファンクションが呼ばれてからのマウスの移動量を得る。

0BH	マウストラップの設定
-----	------------

入 力 AX=0BH

CX=トラップ条件

ES:DX=トラップルーチンアドレス

INT 33H

出 力 なし

解 説 マウス事象によるトラップルーチンを登録する。

<トラップ条件:各ビットが1のときトラップする>

bit0:マウスの移動

bit1:左ボタンの押下

bit2:左ボタンの解放

bit3:右ボタンの押下

bit4:右ボタンの解放

bit5-bit15:0

トラップルーチンはマウス割込みルーチンから、STI 状態になった後 far call で呼ばれる。このときレジスタには次の情報が設定されている。

AX=トラップの要因

bit0:マウスの移動

bit1:左ボタンの押下

bit2:左ボタンの解放

bit3:右ボタンの押下

bit4:右ボタンの解放

BL=左ボタンの状態 (FFH=押/00H=離)
 BH=右ボタンの状態 (FFH=押/00H=離)
 CX=マウスカーソルの水平位置 (0-639)
 DX=マウスカーソルの垂直位置 (0-199 or 0-399)

0CH	マウス移動比の設定
-----	-----------

入 力 AX=0CH

CX=水平マウス移動比

DX=垂直マウス移動比

INT 33H

出 力 なし

解 説 マウスの移動比を設定する。大きいほど動きが遅くなる。0を設定すると自動的に初期値を設定したことになる。

<移動比初期値>

スプライトが 400 ラインのとき 8

スプライトが 200 ラインのとき 16

0DH	マウスの水平移動範囲の設定
-----	---------------

入 力 AX=0DH

CX=マウス移動範囲左 X 座標

DX=マウス移動範囲右 X 座標

INT 33H

出 力 なし

解 説 マウスの水平移動範囲を設定する。初期状態は 0～639。

0EH	マウスの垂直移動範囲の設定
-----	---------------

入 力 AX=0EH

CX=マウス移動範囲上 Y 座標

DX=マウス移動範囲下 Y 座標

INT 33H

出 力 なし

解 説 マウスの垂直移動範囲を設定する。初期状態は 0-199 または 0-399。

0FH	マウスをスリープ状態にする
-----	---------------

入 力 AX=0FH
INT 33H

出 力 なし

解 説 マウス割込みの処理を止め、マウスを使用停止状態にする。マウスカーソルの表示も停止する。スリープ状態からの脱出は、次のファンクション 10H かファンクション 00H で行う。

10H	マウスをアクティブ状態にする
-----	----------------

入 力 AX=10H
INT 33H

出 力 なし

解 説 マウスの使用を再開し、スリープする前の状態に戻す。

11H	マウス状態の取得
-----	----------

入 力 AX=11H
INT 33H

出 力 AX=0000H：マウスはスリープモード
=FFFFH：マウスはアクティブモード

解 説 マウスがスリープかアクティブかを知らせる。

12H	マウスの読出し周期の設定
-----	--------------

入 力 AX=12H
CX=0：8.3ms
=1：16.7ms
INT 33H

出 力 AX=0000H：マウスはスリープモード
=FFFFH：マウスはアクティブモード

解 説 マウスのハードウェア割込みルーチンを呼び出す周期を設定する。初期値は 16.7 ms。

作業領域 034BH : 0000H

マウスの BIOS の作業領域のセグメントは 034BH である。

OFFSET (16 進)	機 能
00-01	マウス移動範囲の左端
02-03	マウス移動範囲の右端
04-05	マウス移動範囲の上端
06-07	マウス移動範囲の下端
08-09	マウス X 座標
0A-0B	マウス Y 座標
0C	マウスカーソル表示フラグ
0D-0E	マウスカーソルセンタ X
0F-10	マウスカーソルセンタ Y
11	以前のボタン状態
12	現在のマウスイベントの状態
13	マウスボタンの状態
14	内部使用
15	マウスアクティブフラグ
16-17	左ボタンを押した回数
18-19	左ボタンを離した回数
1A-1B	右ボタンを押した回数
1C-1D	右ボタンを離した回数
1E-1F	左ボタンを押したときの X 座標
20-21	左ボタンを押したときの Y 座標
22-23	左ボタンを離したときの X 座標
24-25	左ボタンを離したときの Y 座標
26-27	右ボタンを押したときの X 座標
28-29	右ボタンを押したときの Y 座標
2A-2B	右ボタンを離したときの X 座標
2C-2D	右ボタンを離したときの Y 座標
2E-2F	水平方向ミッキーカウント
30-31	垂直方向ミッキーカウント
32-33	水平方向ミッキーカウントの合計
34-35	垂直方向ミッキーカウントの合計
36-37	水平方向の移動比
38-39	垂直方向の移動比
3A-3D	内部使用
3E-3F	マウストラップ条件
40-43	マウストラップアドレス
44-65	内部使用
66	マウスの読出し周期
67	内部使用
68	マウスカーソルの水平サイズ
69	マウスカーソルの垂直サイズ
6A	マウスカーソルのスプライトスイッチ (bit1)
6B	マウスカーソルのスプライトモード (bit2)
6C-6D	マウスカーソルのモノクロ時のカラーコード

6.13 カレンダ時計 BIOS

内蔵クロックチップ μ PD4990AC に日付および時刻の設定・読出しを行う。

●カレンダ時計 BIOS 機能一覧

機能番号 (16 進)	機 能
00	日付の取得
01	日付の設定
02	時刻の取得
03	時刻の設定
04	両面最上部に日付・時刻を表示する
05	画面最上部の日付・時刻表示を終了する
06	画面最上部の日付・時刻表示の状態を得る

●呼出し方

- ①レジスタ／メモリにパラメータ設定
- ②AH ← 機能番号
- ③INT 8CH

00H	日付の取得
-----	-------

入 力 AH=00H
INT 8CH

出 力 CX=年(1980-2079)
DH=月(1-12)
DL=日(1-31)
AL=曜日(0:日 1:月 2:火…6:土)

解 説 現在の日付をクロックチップ μ PD4990AC から読み出す。
AX, CX, DX 以外のレジスタは保持される。

01H	日付の設定
-----	-------

入 力 AH=01H
CX =年(1980-2079)
DH=月(1-12)
DL=日(1-31)
INT 8CH

出力 AL=00：正常終了

 ≡00：パラメータエラー

 bit0：年が異常

 bit1：月が異常

 bit2：日が異常

解説 指定した日付をクロックチップ μ PD4990AC に設定する。
AX, CX, DX 以外のレジスタは保持される。

02H	時刻の取得
-----	-------

入力 AH=02H

INT 8CH

出力 CH=時(0-23)

CL=分(0-59)

DH=秒(0-59)

DL=00

解説 現在の時刻をクロックチップ μ PD4990AC から読出す。
AX, CX, DX 以外のレジスタは保持される。

03H	時刻の設定
-----	-------

入力 AH=03H

CH=時(0-23)

CL=分(0-59)

DH=秒(0-59)

INT 8CH

出力 AL=00：正常終了

 ≡00：パラメータエラー

 bit0：時が異常

 bit1：分が異常

 bit2：秒が異常

解説 指定した時刻をクロックチップ μ PD4990AC に設定する。
AX, CX, DX 以外のレジスタは保持される。

04H	画面最上部に日付・時刻を表示する
-----	------------------

入 力 AH=04H
INT 8CH

出 力 なし

解 説 PC-Engine のコマンドプロセッサが、画面最上部の日付・時刻表示を開始するためのもので、標準的なスクリーンエディタの環境を守っていない場合は使用できない。

05H	画面最上部の日付・時刻表示を終了する
-----	--------------------

入 力 AH=05H
INT 8CH

出 力 なし

解 説 画面最上部の日付・時刻表示を停止するためのもので、アプリケーションソフト起動にこのファンクションで表示が停止される。

06H	画面最上部の日付・時刻表示の状態を得る
-----	---------------------

入 力 AH=06H
INT 8CH

出 力 AL=00：表示中
=FF：表示停止中

解 説 画面最上部の日付・時刻表示の状態を調べる。

6.14 ファンシーフォント BIOS

ファンシーフォント BIOS は、オリジナルフォントの文字サイズ・字体を修飾しながらプリンタ出力用に変換するためのものである。オリジナルフォントの文字サイズは 16×16 ドット(全角)または 16×8 ドット(半角)をサポートする。以下に変換後の字体・サイズの種類を示す。

●字 体

- 1) 袋文字(白抜き文字)
- 2) 斜体
- 3) 強調(太字)
- 4) 影文字
- 5) 1-4 の任意の組合せ

●文字サイズ

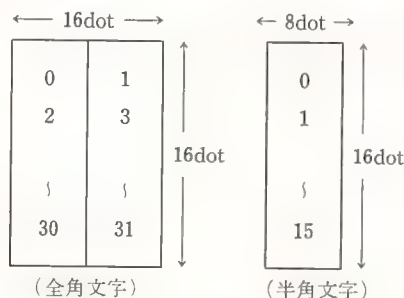
- 1) 16 ドット(24 ドットバッファ)

字体を変えると 16 ドット文字が 16 ドットより大きくなる。このため 24 ドット分のバッファを用意し、そこに格納するモード。24 ドットバッファ中のオリジナルフォントの位置は、上マージン 2 ドット・左マージン 1 ドットである。
- 2) 16 ドット(16 ドットバッファ)

大きくなったフォントをカットして 16 ドット用バッファに詰め込むモード
- 3) 24 ドット(24 ドットバッファ)

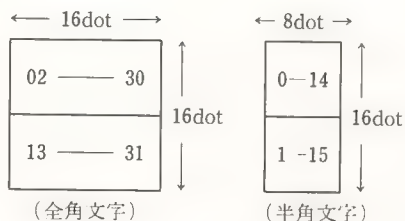
16 ドット文字を 24 ドットに変換後、字体変更して、24 ドット用バッファに詰め込むモード。斜体指定した場合は、24 ドットバッファが横に拡張される。いずれにしろ、上下はマージンカットされる。

●オリジナルフォントバッファの形式

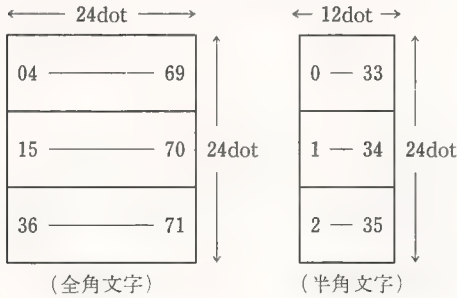


●変換後フォントバッファの形式

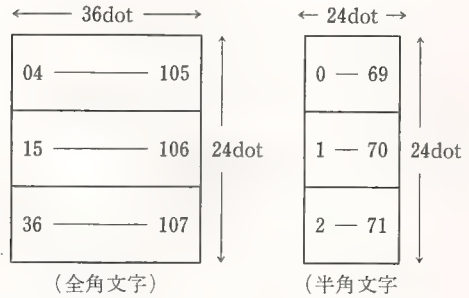
① 16 ドットバッファ



② 24 ドットバッファ・斜体指定なし

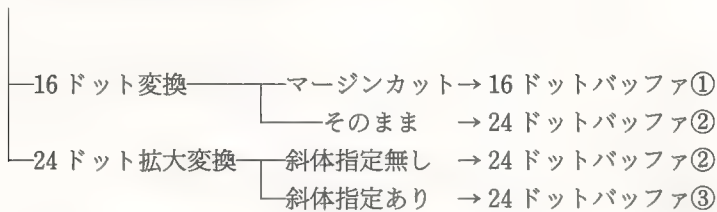


③ 24 ドットバッファ・斜体指定あり



●変換の種類と使用される変換後フォントバッファ

オリジナルフォント(16 ドット)



INT 92H 文字フォントの変換

入 力 CL=変換モード

bit7: 袋文字指定 (1 = 袋文字)

bit6: 斜体指定 (1 = 斜体)

bit5: 強調指定 (1 = 強調)

bit4: 影文字指定 (1 = 影文字)

bit3: 0

bit2: バッファサイズ (1 = 16 ドット / 0 = 24 ドット)

bit1: ドットサイズ (1 = 24 ドット / 0 = 16 ドット)

bit0: 文字サイズ (1 = 半角 / 0 = 全角)

ES: SI=オリジナルフォントバッファのアドレス

ES: DI=変換後のフォントバッファのアドレス

INT 92H

出 力 ES: DI=変換後のフォントバッファ

6.15 スクリーンエディタ BIOS

スクリーンエディタ BIOS は、コマンドプロセッサおよび V3 BASIC における入力をサポートするものである。

●スクリーンエディタ BIOS 機能一覧

機能番号 (16進)	機 能
00	スクリーンエディタの使用開始
01	システムラインの表示・消去
02	スクリーンモードの変更
03	スクロール範囲の変更
04	ヌルキャラクタアトリビュートの変更
05	文字表示(コントロールコード有効)
06	文字表示(コントロールコード無効)
07	1行入力
08	部分行入力

●呼出し方

- ①レジスタ／メモリにパラメータ設定
- ② AH ← 機能番号
- ③ INT 94H

●サポートする画面

桁数：80 または 40

行数：10／20 または 12／25

00H	スクリーンエディタの使用開始
-----	----------------

入 力 AH=00H

AL=画面モード

bit7：システムラインの行数(1=2行／0=1行)

bit6：アトリビュートモード(1=モード1／0=モード0)

bit5-4：行数(11=10行／10=12行／01=20行／00=25行)

bit3：0

bit2：桁数(1=80桁／0=40桁)

bit1：インターレースモード(1=ノンインターレース／0=インターレース)

bit0：0

INT 94H

出 力 なし

解 説 スクリーンエディタ利用時には、必ず最初にこのファンクションを実行しなければならない。このファンクションにより TVRAM 上に次の 2 つのフレームバッファが確保される。

- ・フレームバッファ # 0 (メインスクリーン用)

先頭番地	0000H
フレームバッファの幅	80
フレームバッファの高さ	25

- ・フレームバッファ # 1 (システムライン用)

先頭番地	0FA0H
フレームバッファの幅	80
フレームバッファの高さ	2

最後に AL に従って、画面モード等が設定される。システムラインは、非表示となる。表示する必要がある場合は、この後ファンクション 01H を呼ぶ。

01H	システムラインの表示・消去
-----	---------------

入 力 AH=01H

AL=FFH：システムラインを消去し、全画面をメイン画面とする

≠FFH：システムラインを表示する

SHMD	0	0	0	ファンクションキー表示個数
------	---	---	---	---------------

SHMD= 0：シフトキーを無視する

1：シフト押下でシフト状態の文字列を表示する

ファンクションキー表示個数

0：ファンクションキーを表示しない

5：ファンクションキーを 5 個表示する

10：ファンクションキーを 10 個表示する

INT 94H

出 力 なし

解 説 システムラインの表示・消去を行う。表示時にはファンクションキーの表示個数も指定する。

02H	スクリーンモードの変更
-----	-------------

入 力 AH=02H

AL=画面モード (ファンクション 00H と同じ)

INT 94H

出 力 なし

解 説 スクリーンモードを変更し、画面を初期化する。

03H	スクロール範囲の変更
-----	------------

入 力 AH=03H
 DH=スクロール開始行
 DL=スクロール終了行
 INT 94H

出 力 なし

解 説 メイン画面のスクロール範囲を指定する。

04H	マルチキャラクタアトリビュートの変更
-----	--------------------

入 力 AH=04H
 DX=アトリビュートコード(DH=0にする)
 INT 94H

出 力 なし

解 説 メイン画面のマルチキャラクタで用いるアトリビュートコードを指定する。画面消去・削除等で、マルチキャラクタが使用されるときは、このアトリビュートコードが用いられる。

05H	文字表示(コントロールコード有効)
-----	-------------------

入 力 AH=05H
 DX=2 バイト文字コード
 INT 94H

出 力 なし

解 説 カーソル位置に文字を表示し、カーソルを進める。与えられた文字コードがコントロールコードであった場合には、テキスト BIOS とは異なるスクリーンエディタ特有の処理を行う。それ以外については、テキスト BIOS のファンクション 01H(1文字出力)が呼ばれる。

＜コントロールコードの処理＞

07H	ブザーを鳴らす		
09H	タブ位置までスペースを出力する		
0AH	カーソルを下の行の先頭に移動する		
0BH	カーソルをスクロール範囲の左上隅に移動する		
0CH	スクロール範囲内を消去する		
0DH	カーソルを左端に移動する		
16H	半角入力モードにする	1DH	カーソルを左に移動する
17H	全角入力モードにする	1EH	カーソルを上に移動する
1CH	カーソルを右に移動する	1FH	カーソルを下に移動する

06H	文字表示(コントロールコード無効)
-----	-------------------

入 力 AH=06H

DX=2バイト文字コード

INT 94H

出 力 なし

解 説 /コントロールコードの処理を行わないこと以外は、ファンクション 05H と同じ。コントロールコードは、それに対応した特殊文字を表示する。

07H	1 行入力
-----	-------

入 力 AH=07H

AL=制御パラメータ

ES:BX=入力バッファのアドレス

CX=入力バッファの長さ

DX=待ち時間(0001-7FFFH 0.1 秒単位)

時間制限を設けないときは DX=0

INT 94H

<AL:制御パラメータ>

bit7:最初の挿入・上書モード(1=挿入/0=上書)

bit5:カーソル表示(1=OFF/0=ON)

bit4:カーソルブリンク(1=OFF/0=ON)

bit2: ^T/^N(ヒストリー処理)(1=有り/0=無し)

bit1-0: ^A(HELP 機能)

11=有り(バッファ更新しない)

10=有り(バッファ更新する)

00=なし

出 力 入力バッファ=文字列

AX=終了条件

AH=00H: ^J/^M/^T/^N による終了

=80H: ^C による終了(バッファは更新されない)

=FFH: タイムアウト(バッファは更新されない)

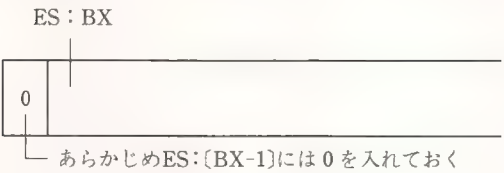
AL MSB: 終了時の挿入・上書(1=挿入(^Jのとき)/0=上書)

bit6-0: 終了時の文字コード

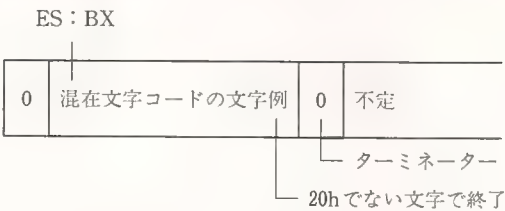
解 説 文字入力を連続して行い、改行キー等が入力された時点で、1行(論理行)を入力バッファに取り込んで戻る。STOP キー等は入力バッファを更新せずに戻る。入力条件として、待ち時間が指定された場合は、待ち時間の間キー入力がない場合、入力バッファを更新せずに戻る。

<入力バッファの形式>

①呼ぶ前



②呼んだ後



<コントロールコードの処理>

コード		機 能
01H	HELP	説明処理のために正常終了として戻る
02H	PRVWD	1つ前のワードに戻る
03H	STOP	中断, 異常処理として戻る
04H	DELWD	カーソル位置から1ワードを削除
05H	ERSLN	カーソル位置からその行の終わりまで消去
06H	NXTWD	1つ先のワードへ進む
07H	BEEP	ブザーを鳴らす
08H	DEL	1文字削除
09H	TAB	タブ位置まで空白を出力する
0AH	LFSPL	ラインフィード/挿入モードで行分割
0BH	HOME	カーソルをホームポジションへ移動
0CH	CLS	スクロール範囲を消去する
0DH	CR	入力終了
12H	INST	挿入モードにする
15H	CANC	1行キャンセル
18H	ENDL	カーソルを行の最後に移動する
1CH	CSRRT	カーソルを右へ移動する
1DH	CSRLT	カーソルを左へ移動する
1EH	CSRUP	カーソルを上へ移動する
1FH	CSRDN	カーソルを下へ移動する
F8H	ROLUP	スクロール範囲をスクロールアップする
F9H	ROLDN	スクロール範囲をスクロールダウンする

08H 部分行入力

入 力 AH=08H

AL=制御パラメータ(ファンクション 07H と同じ)

ES : BX=入力バッファのアドレス

CX=入力バッファの長さ

DX=待ち時間(0001-7FFFH 0.1 秒単位)
時間制限を設けないときは DX=0

INT 94H

出 力 AX=終了条件(ファンクション 07H と同じ)

入力バッファ=文字列

解 説 カーソルが移動した部分だけの文字列を取り込む(V1/V2 BASIC の INPUT 文と同様)以外はファンクション 07H と同じ。

6.16 日本語入力フロントプロセッサ BIOS

日本語入力を行うためのフロントプロセッサ(JFP)を制御する BIOS であり、以下の機能を持つ。

● JFP の機能一覧

機能番号 (16進)	機 能
00	JFP モードの取得
01	JFP モードの設定
02	JFP の初期化
03	JFP の中止
04	第1候補
05	次候補
06	前候補
07	次文節への移動
08	前文節への移動
09	注目文節の延長
0A	注目文節の短縮
0B	文節ひらがな変換
0C	文節カタカナ変換
0D	文節半角変換
0E	全ひらがな変換
0F	全カタカナ変換
10	全半角変換
11	決定
12	ユーザ辞書ファイル名の取得
13	ユーザ辞書ファイル名の設定
14	単語登録
15	単語削除
16	全角半角変換
17	半角全角変換
18	カタカナひらがな変換
19	ひらがなカタカナ変換
1A	シフト JIS → JIS 変換
1B	JIS →シフト JIS 変換
1C	ローマ字変換
1D	画面切変えの通知

●呼出し方

- ① レジスタ／メモリにパラメータ設定
- ② CL ← 機能番号
- ③ INT 8DH

00H	JFP モードの取得
-----	------------

入 力 CL=00H
INT 8DH

出 力 AH=JFP 制御データ
AL=JFP 入力モード
DH=直接入力モード時の最大文字数(16-60)
DL=間接入力モード時の最大文字数(16-30)

〈JFP 制御データ〉

bit 7:1=JFP OFF/0=JFP ON
bit 6:1=全角半角切替え禁止/0=許可
bit 5-4:予約
bit 3:1=外字作成禁止/0=許可
bit 2:1=単語削除禁止/0=許可
bit 1:1=単語登録禁止/0=許可
bit 0:1=画面切替え禁止/0=許可

〈JFP 入力モード〉

bit 7:1=直接入力モード/0=間接入力モード
bit 6-3:1=予約
bit 2:1=カタカナ/0=ひらがな
bit 1:1=かなモード/0=英字モード
bit 0:1=全角/0=半角

解 説 JFP の各種モードを取得する。

01H	JFP モードの設定
-----	------------

入 力 CL=01H
AH=JFP 制御データ(ファンクション 00H と同じ)
AL=JFP 入力モード(ファンクション 00H と同じ)
DH=直接入力モード時の最大文字数(16-60)
DL=間接入力モード時の最大文字数(16-30)
INT 8DH

出 力 なし

解 説 JFP の各種モードを設定し、モードにしたがって再初期化する。

02H	JFP の初期化
-----	----------

入 力 CL=02H
INT 8DH

出 力 なし

解 説 JFP の各種モードを初期設定値に戻す。

- 1) 直接入力モード時の最大文字数=60
- 2) 間接入力モード時の最大文字数=30
- 3) JFP バッファのクリア
- 4) 半角英数モードの選択
- 5) 単語登録・削除の許可
- 6) 外字作成の許可
- 7) JFP ON
- 8) デフォルトユーザ辞書のロード
- 9) デフォルト外字ファイルのロード

03H	JFP の中止
-----	---------

入 力 CL=03H
INT 8DH

出 力 なし

解 説 JFP の実行中(未決定状態・単語登録・外字作成など)から強制的に抜け出す。

04H	第 1 候補
-----	--------

入 力 CL=04H
DS:SI=全角の読み文字列
DS:DI=出力バッファ
INT 8DH

出 力 AL=注目文節先頭位置
AH=文節長

解 説 読み文字列の第 1 候補を返す。このファンクションは他の変換ファンクションを使用する前に実行する。

05H	次候補
-----	-----

入 力 CL=05H
 DS:SI=全角の読み文字列
 DS:DI=出力バッファ
 INT 8DH

出 力 AL=注目文節先頭位置
 AH=文節長

解 説 注目文節の次候補を返す。

06H	前候補
-----	-----

入 力 CL=06H
 DS:SI=全角の読み文字列
 DS:DI=出力バッファ
 INT 8DH

出 力 AL=注目文節先頭位置
 AH=文節長

解 説 注目文節の前候補を返す。

07H	次文節への移動
-----	---------

入 力 CL=07H
 DS:SI=全角の読み文字列
 DS:DI=出力バッファ
 INT 8DH

出 力 AL=注目文節先頭位置
 AH=文節長

解 説 注目文節を次の(右の)文節へ移動する。現在の文節が最後の場合は、先頭文節へ移る。

08H	前文節への移動
-----	---------

入 力 CL=08H
 DS:SI=全角の読み文字列
 DS:DI=出力バッファ
 INT 8DH

出 力 AL=注目文節先頭位置
 AH=文節長

解 説 注目文節を前の(左の)文節へ移動する。現在の文節が先頭の場合は、文節は移動しない。

09H	注目文節の延長
-----	---------

入 力 CL=09H
 DS:SI=全角の読み文字列
 DS:DI=出力バッファ
 INT 8DH

出 力 AL=注目文節先頭位置
 AH=文節長

解 説 注目文節を延長する。先頭位置は変化しない。延長された注目文節は直ちに再変換される。

0AH	注目文節の短縮
-----	---------

入 力 CL=0AH
 DS:SI=全角の読み文字列
 DS:DI=出力バッファ
 INT 8DH

出 力 AL=注目文節先頭位置
 AH=文節長

解 説 注目文節を短縮する。先頭位置は変化しない。短縮された注目文節は直ちに再変換される。

0BH	文節ひらがな変換
-----	----------

入 力 CL=0BH
 DS:SI=全角の読み文字列
 DS:DI=出力バッファ
 INT 8DH

出 力 AL=注目文節先頭位置
 AH=文節長

解 説 注目文節をひらがな変換する。

0CH	文節カタカナ変換
-----	----------

入 力 CL=0CH
 DS:SI=全角の読み文字列
 DS:DI=出力バッファ
 INT 8DH

出 力 AL=注目文節先頭位置
AH=文節長

解 説 注目文節をカタカナ変換する。

0DH	文節半角変換
-----	--------

入 力 CL=0DH
DS:SI=全角の読み文字列
DS:DI=出力バッファ
INT 8DH

出 力 AL=注目文節先頭位置
AH=文節長

解 説 注目文節を半角に変換する。

0EH	全ひらがな変換
-----	---------

入 力 CL=0EH
DS:SI=全角の読み文字列
DS:DI=出力バッファ
INT 8DH

出 力 AL=0
AH=文長

解 説 読み文字列全部をひらがなに変換する。非かな文字は影響を受けない。

0FH	全カタカナ変換
-----	---------

入 力 CL=0FH
DS:SI=全角の読み文字列
DS:DI=出力バッファ
INT 8DH

出 力 AL=0
AH=文長

解 説 読み文字列全部をカタカナに変換する。非かな文字は影響を受けない。

10H	全半角変換
-----	-------

入 力 CL=10H
DS:SI=全角の読み文字列
DS:DI=出力バッファ
INT 8DH

出 力 AL=0
AH=文長

解 説 読み文字列全部を半角に変換する。

11H	決定
-----	----

入 力 CL=11H
INT 8DH

出 力 なし

解 説 変換結果を学習し、RAM CG の学習領域を更新する。

12H	ユーザ辞書ファイル名の取得
-----	---------------

入 力 CL=12H
DS:SI=出力バッファ(20 バイト)
INT 8DH

出 力 出力バッファ=ファイル名

解 説 ユーザ辞書のファイル名(ASCIZ)を得る。

13H	ユーザ辞書ファイル名の設定
-----	---------------

入 力 CL=13H
DS:DI=ユーザ辞書ファイル名(ASCIZ:20 バイト以内)
INT 8DH

出 力 AX=0:正常終了
 ≠0:エラー(AH=不定)
 AL=3:致命的ディスクエラー
 =4:ユーザ辞書が大き過ぎる

解 説 ユーザ辞書のファイル名を設定する。辞書のロードは行われない。

14H	単語登録
-----	------

入 力 CL=14H

AX=品詞情報

DS:SI=全角の読み文字列

DS:DI=全角かな・漢字文字列(登録する単語)

INT 8DH

出 力 AX=0:正常終了

≠0:エラー(AH=不定)

AL=2:単語がすでに存在する

=3:致命的ディスクエラー

=4:ユーザ辞書が大き過ぎる

=5:同音義語が多過ぎる

解 説 ユーザ辞書に単語登録する。読みとしては全角ひらがな・カタカナを使用する。

<品詞情報>

bit 15:名詞フラグ(1=名詞)

bit 14-11:名詞の種類

bit 14:1=「的」が接続する名詞

bit 13:1=「性」が接続する名詞

bit 12:1=役職・人名

bit 11:1=地名

bit 10:1=接頭語「御」が接続する

bit 9-7:000

bit 6:形容動詞フラグ

bit 5:副詞フラグ

bit 4:形容詞フラグ

bit 3-0:動詞タイプ

0000:動詞でない

0001:カ行5段

0010:ガ行5段

0011:サ行5段

0100:タ行5段

0101:ナ行5段

0110:ハ行5段

0111:マ行5段

1000:ラ行5段

1001:アワ行5段

1010:1段

1011:サ変

1100:ザ変

1101:カ変

15H	単語削除
-----	------

入 力 CL=15H

AX=品詞情報

DS:SI=全角の読み文字列(削除する読み)

DS:DI=全角かな・漢字文字列(削除する単語)

INT 8DH

出力 AX=0：正常終了
 ≠0：エラー(AH=不定)
 AL=2：単語が存在しない
 =3：致命的ディスクエラー

解説 ユーザ辞書から単語を削除する。

16H	全角半角変換
-----	--------

入力 CL=16H
 DS：SI=全角文字列(シフト JIS)
 DS：DI=半角出力バッファ
 INT 8DH

出力 AX=0：正常終了
 ≠0：エラー(半角に変換できなかった位置。1=先頭)

解説 全角文字列を半角文字列に変換する

17H	半角全角変換
-----	--------

入力 CL=17H
 DS：SI=半角文字列
 DS：DI=全角出力バッファ(シフト JIS)
 INT 8DH

出力 AX=0：正常終了
 ≠0：エラー(半角に変換できなかった位置。1=先頭)

解説 半角文字列を全角文字列に変換する。2 バイト半角文字(JIS：29xx-2Bxx)は使用できない。

18H	カタカナひらがな変換
-----	------------

入力 CL=18H
 DS：SI=全角カタカナ文字列
 DS：DI=全角ひらがな文字列
 INT 8DH

出力 AX=0：正常終了
 ≠0：エラー(変換できなかった位置。1=先頭)

解説 カタカナ文字列をひらがな文字列に変換する。

19H	ひらがなカタカナ変換
-----	------------

入 力 CL=19H

DS:SI=全角ひらがな文字列

DS:DI=全角カタカナ文字列

INT 8DH

出 力 AX=0:正常終了

≠0:エラー(変換できなかった位置。1=先頭)

解 説 ひらがな文字列をカタカナ文字列に変換する。

1AH	シフト JIS → JIS 変換
-----	------------------

入 力 CL=1AH

AX=シフト JIS コード

INT 8DH

出 力 AX=JIS コード

=FFFFH:無効なシフト JIS コード

解 説 シフト JIS コードを→ JIS コードに変換する。

〈有効なシフト JIS コードの範囲〉

1) 第1バイト:81-9FH or E0-FCH

2) 第2バイト:40-FCH

1BH	JIS →シフト JIS 変換
-----	-----------------

入 力 CL=1BH

AX=JIS コード

INT 8DH

出 力 AX=シフト JIS コード

=FFFFH:無効な JIS コード

解 説 JIS コードを→シフト JIS コードに変換する。

〈有効な JIS コードの範囲〉

1) 第1バイト:21-7EH

2) 第2バイト:21-7EH

1CH	ローマ字変換
-----	--------

入 力 CL=1CH
DS:SI=半角アルファベット文字列
DS:DI=出力バッファ(全角ひらがな文字列)
INT 8DH

出 力 AX=0:正常終了
≠0:エラー(変換できなかった位置。1=先頭)

解 説 アルファベット文字列をローマ字変換してひらがな文字列にする。

1DH	画面切換えの通知
-----	----------

入 力 CL=1DH
AL=切換え要因
bit 7-4:0000
bit 3:フレームバッファ設定ファンクション
bit 2:表示画面設定ファンクション
bit 1:表示モード設定ファンクション
bit 0:ファンクションキー表示ファンクション
INT 8DH

出 力 AX=0:正常終了
≠0:エラー(変換できなかった位置。1=先頭)

解 説 テキスト BIOS の AL で示されるファンクションが呼ばれたときにこのファンクションを用いて JFP に通知する。JFP はこれらの要因別に必要な処理をする。

作業領域	0440H:0000H
------	-------------

JFP の作業領域のセグメントは 0440H である。

OFFSET (16 進)	機 能
00-03	変換入力バッファのアドレス
04-07	変換出力バッファのアドレス
08-0B	ステータス領域のアドレス
0C-0F	ユーザ辞書環境のアドレス
10	JFP 入力モード
11	JFP 制御データ
12	全角モード制御データ

6.17 数値演算 BIOS

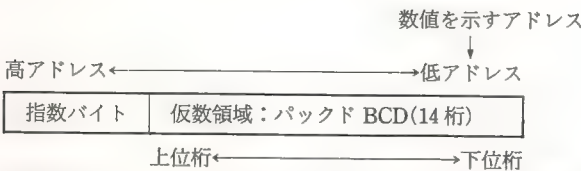
数値演算 BIOS は、倍精度型の 10 進浮動小数点数値に対して、四則演算・比較演算・関数演算・16 ビット 2 進数との相互変換・ASCII 入出力を行うものである。

●数値演算 BIOS 機能一覧

割り込み番号	機 能
INT A0H	倍精度加算
INT A1H	倍精度減算
INT A2H	倍精度乗算
INT A3H	倍精度除算
INT A4H	倍精度比較
INT A5H	整数変換
INT A6H	実数変換
INT A7H	ASCII 入力
INT A8H	ASCII 出力
INT B0H	SIN 関数
INT B1H	COS 関数
INT B2H	TAN 関数
INT B3H	ARCTAN 関数
INT B4H	指数関数
INT B5H	対数関数
INT B6H	べき乗
INT B7H	平方根

●数値表現

入出力に用いられる数値(変数)は 8 バイトからなり、有効数字 14 桁の浮動小数点表現を採用している。なお演算パッケージ内部では、16 桁のメモリ・アキュムレータを用いて計算している。



仮数領域： 7 バイトからなり、14 桁のパックド BCD を構成する。数値の仮数部の絶対値を保持する。仮想小数点は最上位桁のすぐ上にあるものとし、ゼロ以外の全ての数値データは、 $0.1 \leq$ 仮数部 < 1 となるように正規化されなくてはならない。例えば、 $1E+00$ や $0.01E+02$ のような表現は、すべて $0.1E+02$ の形に修正する。

指数バイト： 1 バイトで仮数部の符号および指数部を保持する。仮数部の符号は指数バイトの MSB で表し、0=正/1=負である。

指数バイトのビット 6-0 は数値の指数部を表し、2 の補数表現を用いた 7 ビットの 2 進数に 40H のパイアスを加えた形式を採用している。したがってその符号ビット(ビット 6)は、0=負/1=正となる。

ゼロの表現：指数部バイトをゼロクリアすることで表現する

表現可能な数値： $1E-64 \leq ABS < 1E+63$ (ABS は数値の絶対値)

●演算結果フラグ

当演算パッケージは、オーバーフロー／アンダーフロー等の演算結果の状態を、OF, SF, ZF, CF の 4 つのフラグの組合せで示す。これらのフラグの指定状態は、各演算ルーチンを通じて共通である。ただし倍精度比較演算は例外であり、比較結果そのものをフラグに出力する。これらのフラグの組合せとその表す状態の一覧を示す。

CF	SF	ZF	OF	意味
0	S	0	0	正常終了
0	0	1	0	演算結果ゼロ
0	S	0	1	アンダーフロー
1	0	0	0	シンタックスエラー
1	0	1	1	0 除算エラー
1	S	0	1	オーバーフロー
1	S	0	0	計算不能

S : 0=正/1=負

正常終了：

演算結果の符号を SF に設定して戻る。

演算結果ゼロ：

演算結果が厳密にゼロになった場合には、出力変数の指数部バイトをゼロクリアし、ZF に 1 を立てて戻る。

アンダーフロー：

演算結果の絶対値が $1E-64$ 未満となった場合には、出力変数の指数部バイトをゼロクリアし、OF に 1 を立てて戻る。また、演算結果の符号を SF に設定する。関数演算の場合には、入力値を演算した結果がとるべき符号を SF に設定する。

シンタックスエラー：

ASCII 入力ファンクションにおいて ASCII 数値表現が正しくなかった場合、もしくは ASCII 出力ファンクションにおいてパラメータの指定に誤りがあった場合には、CF に 1 を立てて戻る。

0 除算エラー：

- 1) 0 で除算した
- 2) TAN 演算において COS 演算がゼロになるような値が入力された
- 3) べき乗演算において底が 0 で指数が負である演算を行おうとした

これらの場合には、出力変数に許容最大値 ($\pm 0.9999999999999999E+63$) を設定し、CF, OF, ZF に 1 を立てて戻る。このときの最大許容値の符号は演算前のものを採用する。

オーバーフロー：

演算結果の絶対値が $1E+63$ 以上となった場合には、出力変数に許容最大値を設定し、CF, OF に 1 をたてて戻る。このときの最大許容値の符号はオーバーフロー発生時のものを採用する。関数演算の場合には入力値を演算した結果がとるべき符号を設定する。ただし、SIN・COS・TAN 演算の場合には入力時のものをを用いる。また、この符号を SF にもコピーする。

計算不能：

- 1) SQRT・LOG 演算で引数 < 0
- 2) べき乗演算で底 < 0 かつ指数が整数でない

このとき CF, SF に 1 を立てて戻る。入力変数の値は保存される。また引数(底)の符号を SF にコピーする。

●注意

各演算ともに入力変数のデータ・フォーマットが正しくない場合には、動作は保証されない。

INT A0H -A3H	倍精度四則演算
-----------------	---------

入 力 DS:DI=被演算数のアドレス
DS:SI=演算数のアドレス
INT A0H-A3H
INT A0H:加算
INT A1H:減算
INT A2H:乗算
INT A3H:除算

出 力 DS:DI が示す変数=演算結果(数値)
Flags(CF, SF, ZF, OF)=演算結果ステータス

解 説 倍精度の四則演算を計算し、被演算数に格納する。演算数の方は破壊されない。
 $[DI] = [DI] \bigcirc [SI]$

INT A4H	倍精度比較演算
---------	---------

入 力 DS:DI=被減算数のアドレス
DS:SI=減算数のアドレス
INT A4H

出 力 SF, ZF=比較結果
CF, OF=0

解 説 倍精度の比較を行い、フラグをセットする。変数は破壊されない。
 $[DI] - [SI]$

INT A5H	整数変換
---------	------

入 力 DS:DI=変換する変数のアドレス
INT A5H

出 力 AX=符号付き2進整数
Flags(CF, SF, ZF, OF)=演算結果ステータス

解 説 倍精度変数を16ビットの符号付き2進整数(2の補数)に変換する。整数の取り得る値は、 $-32768 \leq AX \leq 32767$ であり、それを越える値についてはオーバーフローとなり、AXには-32768ないし32767が入る。変換する変数の内容は変化しない。

INT A6H	実数変換
---------	-------------

入 力 AX=符号付き 2 進整数

DS:DI=変換後の倍精度変数を格納するアドレス

INT A6H

出 力 DS:DI が示す変数=変換結果

Flags(CF, SF, ZF, OF)=正常終了またはゼロ演算結果のみ

解 説 16 ビットの符号付き 2 進整数(2つの補数表現)を倍精度実数に変換する。

INT A7H	ASCII 入力
---------	-----------------

入 力 DS:SI=ASCII 表現の文字列の先頭アドレス

DS:DI=変換後の倍精度変数を格納するアドレス

INT A7H

出 力 DS:DI が示す変数=変換結果

DS:SI=ASCII 数値の終了アドレス+1

Flags(CF, SF, ZF, OF)=変換結果ステータス

解 説 ASCII 表現の数値を倍精度実数に変換する。ASCII 数値の形式は固定表現、浮動表現を問わない。変換不能の場合でも[DS:DI]の変数の内容は破壊される。

INT A8H	ASCII 出力
---------	-----------------

入 力 DS:SI=出力する倍精度変数のアドレス

DS:BX=変換後の ASCII 数値を格納する領域の先頭アドレス

AL=出力フォーマット

CH=整数部の長さ

CL=小数部の長さ(小数点を含む)

ただし、AL に固定フォーマットを指定した場合には

CH+CL<250 になる値を指定すること。

INT A8H

出 力 DS:BX が示す領域=ASCII 数値

CX=ASCII 数値の長さ

Flags(CF, SF, ZF, OF)=変換結果ステータス

解 説 倍精度実数を出力フォーマットの指定に従って ASCII 数値に変換する。出力時の ASCII 数値の終端には NULL(00H)が入る。[DS:DI]の変数の内容は変化しない。

- 〈AL：出力フォーマット〉
- bit 7：1＝固定フォーマットによる出力指定
ビット 6～1 および CH，CL の指定が有効になる。
0＝自由フォーマットによる出力指定
14 桁に納まる場合は固定小数点表現，納まらない場合は，浮動小数点表現が採用される。ビット 6～1 および CH，CL の指定は無効になる。
- bit 6：1＝浮動小数点表現による出力指定
コンマ出力(bit 1)の指定は無効になる
指数部の長さ(＝4)を，CH・CL のいずれかに含めること
0＝固定小数点表現による出力指定
- bit 5：1＝符号を数値の右側に出力する
0＝符号を数値の左側に出力する
- bit 4：1＝正号に‘+’を指定する
0＝正号に空白(‘ ’ or ‘*’)を指定する
- bit 3：1＝空白部に‘*’を指定する
0＝空白部に‘ ’を指定する
- bit 2：1＝数値の左側に‘¥’を出力する
0＝‘¥’は出力しない
- bit 1：1＝整数部を 3 桁ごとにコンマで区切って出力する
0＝コンマは出力しない
- bit 0：未使用(0 にしておく)

INT B0H	SIN 関数
---------	--------

- 入 力 DS：DI＝倍精度変数
INT B0H
- 出 力 DS：DI が示す変数＝関数値
Flags(CF, SF, ZF, OF)＝演算結果ステータス
- 解 説 単位をラジアンとした場合の SIN 関数の値を返す。
[DI]=SIN([DI])

INT B1H	COS 関数
---------	--------

- 入 力 DS：SI＝倍精度変数
INT B1H
- 出 力 DS：DI が示す変数＝関数値
Flags(CF, SF, ZF, OF)＝演算結果ステータス
- 解 説 単位をラジアンとした場合の COS 関数の値を返す。
[DI]=COS([DI])

INT B2H	TAN 関数
---------	--------

入 力 DS : DI=倍精度変数
INT B2H

出 力 DS : DI が示す変数=関数値
Flags(CF, SF, ZF, OF)=演算結果ステータス

解 説 単位をラジアンとした場合の TAN 関数の値を返す。
[DI]=TAN([DI])

INT B3H	ARCTAN 関数
---------	-----------

入 力 DS : DI=倍精度変数
INT B3H

出 力 DS : DI が示す変数=関数値
Flags(CF, SF, ZF, OF)=演算結果ステータス

解 説 ARCTAN 関数の値を単位をラジアンとして返す。
[DI]=ARCTAN([DI])

INT B4H	指数関数
---------	------

入 力 DS : DI=倍精度変数
INT B4H

出 力 DS : DI が示す変数=関数値
Flags(CF, SF, ZF, OF)=演算結果ステータス

解 説 e を底とする指数関数の値を返す。
[DI]=EXP([DI])

INT B5H	対数関数
---------	------

入 力 DS : DI=倍精度変数
INT B5H

出 力 DS : DI が示す変数=関数値
Flags(CF, SF, ZF, OF)=演算結果ステータス

解 説 e を底とする対数関数(自然対数)の値を返す。
[DI]=LN([DI])

INT B6H	べき乗
---------	-----

入 力 DS : DI = 倍精度変数(底)
 DS : SI = 倍精度変数(指数)
 INT B6H

出 力 DS : DI が示す変数 = 演算結果
 Flags(CF, SF, ZF, OF) = 演算結果ステータス

解 説 べき乗を計算する。底が負のときは指数は整数でなければならない。
 $[DI] = [DI]^{[SI]}$

INT B7H	平方根
---------	-----

入 力 DS : DI = 倍精度変数
 INT B7H

出 力 DS : DI が示す変数 = 演算結果
 Flags(CF, SF, ZF, OF) = 演算結果ステータス

解 説 平方根を計算する。
 $[DI] = \text{SQR}([DI])$

6.18 出力専用ポート保存領域

出力専用のポートへ出力した値のコピーをこの領域に格納する。これは BIOS の BOOK KEEPING 領域であると同時に、アプリケーションソフトと BIOS でポートの整合性をとるためのものである。したがって、アプリケーションソフトがこれらのポートを変更した場合には、この領域の値も必ず更新しなくてはならない。

●ポート保存領域：セグメント 40H

オフセット	名 前	コピーするポート
0000 : 2 Byte	ColComp	port 106H
0002 : 2 Byte	RGBComp	port 108H
0004 : 2 Byte	MskMode	port 10AH
0006 : 2 Byte	PalMode	port 10CH
0008 : 2 Byte	DropCol	port 10EH
000A : 2 Byte	PageMsk	port 110H
000C : 2 Byte	XparG0	port 124H
000E : 2 Byte	XparG1	port 126H
0010 : 2 Byte	XparTxt	port 12EH
0012 : 2 Byte	MskLeft	port 130H
0014 : 2 Byte	MskRit	port 132H
0016 : 2 Byte	MskTop	port 134H
0018 : 2 Byte	MskBtm	port 136H
001A : 2 Byte	TxtMode	port 148H
001C : 32Byte	Palsave	port 300H to 33FH
005C : 2 Byte	VDPCtl	port 504H
005E : 1 Byte	ArtMode	port 21H (8251 mode)
005F : 1 Byte	ArtCmd	port 21H (8251 command)
0060 : 1 Byte	Text8	port 30H
0061 : 1 Byte	TodCtl	port 40H
0062 : 1 Byte	KeyMod1	port 197H (1)
0063 : 1 Byte	KeyMod2	port 197H (2)
0064 : 1 Byte	DskMode	port 1B0H
0065 : 1 Byte	DskCtl	port 1B2H
0066 : 1 Byte	MtrCtl	port 1B4H
0067 : 1 Byte	DskMisc	port 1B6H
0068 : 2 Byte	BusyFlg	BIOS busy bits
006A : 2 Byte	GrRes	port 102H

※ポート 102H は IN 可能だが、ポートのコピーを持つ必要がある。

※ BusyElg は BIOS が使用中かどうかを示すフラグであり、ポートのコピーではない。

第7章 ファイルシステム

PC-Engine ファイルシステムは PC-Engine の DOS 機能を担当するものであり、その機能には次のものがある。

- ファイルアクセス
 - ハンドルを用いたファイルアクセス
 - デバイスドライバによる I/O アクセス
 - MS-DOS 形式と N88-BASIC 形式の 2 つのディスクフォーマット
- プログラムのロード&ゴー
 - メモリ管理下のプログラムロード
- メモリ管理
 - メモリブロックの作成／削除／変更
- デバイスドライバ
 - BIOS を使用した I/O 処理
 - ブロックデバイス（ディスク）
 - ESC シーケンスの処理
- 割込みルーチン管理
 - 割込みベクタの設定／削除／変更

7.1 システムインターラプトコール

PC-Engine では INT 23H-INT 26H, INT 9EH, INT 9FH が下記の目的で使用されている。

割込み番号	機 能
INT 23H	CTRL-C ブレーク処理
INT 24H	ディスクエラーハンドラ
INT 25H	ディスクセクタリード
INT 26H	ディスクセクタライト
INT 9EH	FCB ファイルネームの設定
INT 9FH	内部コマンドの実行

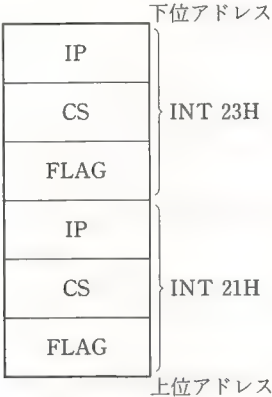
INT 23H	CTRL-C ブレーク処理
---------	---------------

入 力 なし

出 力 なし

解 説 INT 21H ファンクションコールの実行時に CTRL-C が入力され、かつ CTRL-C 検査モードが選択されている場合、システムは INT 23H の割込みを起動する。この割込みルーチンは初期状態では実行中のプロセスを中断し、シェルのコマンド受け付けルーチンに制御を戻すようになっている。

INT 23H の割込みベクタは、プロセス起動時にプロセスコントローラにコピーされ、終了時に元の値に戻されるため、このベクタをユーザが書き換え、プロセス自身の INT 23H ルーチンを設けることができる。このとき、CTRL-C にもかかわらず処理を継続させたい場合には、入力時の全てのレジスタを再設定し、IRET で抜ける必要がある。なお、入力時の各レジスタの値は、ファンクションコールを発行したときと同じである。またスタック状態は以下のようになっているため、INT 23H のスタックを捨ててから IRET を行うことにより、直後ファンクションコールの次のアドレスに移行することができる。



CTRL-C チェックはファンクションの頭で1回だけ行われるため、長い処理時間を要するようなファンクションの場合でも、その処理の途中でブレークをかけることはできない。またファンクション 33H により CTRL-C のチェックが禁止されている場合、INT 23H の呼出しは一切行われない。

INT 24H	ディスクエラーハンドラ
---------	-------------

入 力 AH=エラー環境

bit1= 0 : リードエラー

= 1 : ライトエラー

AL = エラー発生時のドライブ

DI = エラーコード

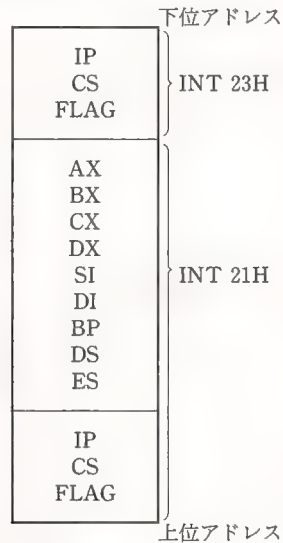
SI = エラー発生時のドライバオフセットアドレス

BP = エラー発生時のドライバセグメントアドレス

出力 AL=ユーザ処理情報

- 00：エラーを無視し、ユーザプログラムに制御を戻す。
- 01：エラーが生じた時のドライバの処理を再試行する。
- 02：現在のプロセスを中止し上位プロセスに制御を戻す。

解説 ディスクアクセス中に回復不能なエラーが検出された場合に呼び出される。初期状態では、この割込みルーチンはエラーの種類を画面に表示し、プロセス中止／アクセス再試行／エラー無視のいずれかをユーザに選択させるようになっている。この選択情報は IRET 時にレジスタ AL に格納され、システムはこの値に基づいて後続の処理を行う。INT 24H の割込みベクタは、プロセス起動時にプロセスコントローラにコピーされ、終了時に元の値に戻されるため、このベクタをユーザが書き換えて、プロセス自身の INT 24H ルーチンを設けることができる。このとき、入力時のレジスタの保存は必要ない。また呼び出された時のスタック状態は以下のようになっているため、INT 24H のスタックと INT 21H のレジスタをポップして IRET を行くと、直接ファンクションコールの次のアドレスに移行することができる。

**<エラーコード：DI レジスタ>**

- 00H：プロテクトされたディスクへの書込み
- 01H：存在しないローカル・ユニット番号を指定
- 02H：ドライブが非レディ状態
- 03H：不正なコマンド番号
- 04H：FDC の CRC エラー
- 05H：コマンドパラメータ領域の長さが不適正
- 06H：FDC のシークエラー
- 07H：不正なメディアタイプ
- 08H：セクタが検出不能
- 0AH：書込み時のエラー
- 0BH：読出し時のエラー
- 0CH：不定エラー

INT 25H/26H	ディスクセクタリード／ライト
-------------	----------------

入 力 AL=ドライブ番号 (0 : A, 1 : B, ……)

CX=アクセスセクタ数

DX=開始セクタ番号

DS : BX=リード／ライトバッファアドレス

出 力 CF= 0 : 正常終了

= 1 : エラー

AL=エラーコード (INT 24H と同じ)

DI=リード／ライトに成功したセクタ数

CX=リード／ライトに失敗したセクタ数

解 説 ディスクに対し、セクタ単位でのリード／ライト処理を行う。このルーチンは、RETF 命令を用いてユーザプログラムに復帰してくるため、ユーザ側でスタック上に積まれたフラグをポップしてやる必要がある。エラーが発生した場合には INT 24H のエラー処理は起動せず、単に CF を 1 にして戻ってくる。

指定したセクタの範囲がディスクの最終セクタを超えた場合、要求されたセクタ数の全てがアクセスできなくても、CF に 1 を返さないことがある。この結果は使用しているデバイスドライバに依存し、標準のフロッピーディスクドライバでは、最終セクタを超えてもエラーは返さない。

INT 9EH	FCB ファイルネームの設定
---------	----------------

入 力 DS : SI=コマンドラインの格納開始アドレス

ES : DI=FCB のファイルネームフィールドの開始アドレス

出 力 AH = 00H : 最初のパス名のドライブ名は正常

= FFH : 最初のパス名のドライブ名が無効

AL = 00H : 2 番目のパス名のドライブ名は正常

= FFH : 2 番目のパス名のドライブ名が無効

解 説 ASICZ 文字列で与えられたコマンドラインを解析し、FCB のファイルネームフィールドに格納する。

〈FCB のファイルネームフィールド : 32 バイト〉

FCB1 : DB ? ; ドライブ番号 (デフォルト : 0, A : 1, …)

DB 11 DUP(?) ; ファイル名

DB 4 DUP(?) ; システム予約

FCB 2 : DB ? ; ドライブ番号

DB 11 DUP(?) ; ファイル名

DB 4 DUP(?) ; システム予約

INT 9FH	内部コマンドの実行
---------	-----------

入 力 DS:SI=コマンドライン (ASCIZ 文字列) の格納開始アドレス

BX=作業領域の開始セグメントアドレス

CX=作業領域のパラグラフサイズ

AL=コマンド動作モード

bit 7-3: 00000

bit 2 : 0=キー入力待ちを行う/1=行わない

bit 1 : 0=文字表示を行う/1=行わない

bit 0 : 1

出 力 CF = 0 : 正常終了

= 1 : エラー

AX = エラーコード

解 説 PC-Engine の内部コマンドを (アプリケーションから) 実行する。文字表示は全てスクリーンエディタに対して行われるため、スクリーンエディタの環境にない場合は、AL=07H で呼び出さなくてはならない。作業領域は最低でも 2K バイトは必要となる。コマンドによっては作業領域が不足して実行できない場合もある。コピー系の命令 (DISK-COPY, COPY 等) は作業領域が少ないと非常に時間がかかることがある。セグメントレジスタおよび IP, SP レジスタ以外はすべて破壊される。

〈エラーコード〉

0 : 作業領域が足りない

1 : メモリアロケーションが壊れている

2 : 指定されたドライブは存在しない

3 : 内部コマンドではない

4 : 日付の設定が間違っている

5 : 時刻の設定が間違っている

6 : オプションスイッチの設定が間違っている

7 : オプションスイッチは使えない

8 : COLOR コマンドのパラメータがない

9 : 色コードが間違っている

10 : 文字アトリビュートコードが間違っている

11 : ON/OFF 以外のパラメータは使用できない

12 : 指定されたディレクトリは使えない

13 : 指定されたファイルはない

14 : ディスクが一杯である

15 : ディレクトリ名として使えない文字がある

16 : 既にディレクトリがあり MKDIR できない

17 : パス名として使えない文字がある

18 : MKDIR コマンドのパラメータがない

19 : RMDIR コマンドのパラメータがない

- 20: 既にファイルが存在するので RMDIR できない
- 21: 指定されたディレクトリはカレントパス上にあるので RMDIR できない
- 22: 指定されたファイルが見つからない
- 23: ファイル名として使えない文字がある
- 24: 指定されたファイルが読めない
- 25: DELETE できるファイルはない
- 26: RENAME するファイルが指定されていない
- 27: ディレクトリを RENAME することはできない
- 28: ファイル名をディレクトリ名に RENAME できない
- 29: 同じファイル名が既に存在するので RENAME できない
- 30: 異なるドライブ間では RENAME できない
- 31: ファイル名の指定がない
- 32: ファイル名として使えない文字がある
- 33: 同じディレクトリ同士の RENAME はできない
- 34: ファイルに書込みができない
- 35: ファイルが読めない
- 36: ファイルに書込みができない
- 37: ディスクが一杯である
- 40: フォーマットできるドライブではない
- 41: フォーマットに失敗した
- 42: フォーマットに必要なメモリが足りない
- 43: DISKCOPY, SYS コマンドでドライブ名がない
- 44: DISKCOPY は同一ドライブ間ではできない
- 45: 種類の異なるフロッピーでは DISKCOPY できない
- 46: ドライブが読めない
- 47: ドライブに書込みができない
- 52: EDIT コマンドでファイルが修正できない
- 54: ヘルプメッセージがない
- 65: SYS, FORMAT/S コマンドで IPL が読めない
- 66: SYS, FORMAT/S コマンドで IPL が書き込めない
- 67: カレントドライブにシステムディスクがない
- 68: ボリュームラベルとして使えない文字がある
- 69: ボリュームラベルが書き込めない
- 71: ファイルが大きすぎて EDIT コマンドは使えない
- 72: EDIT コマンドでファイルの更新ができなかった(ファイルのライトプロテクト等)

7.2 プロセスコントローラ

この領域は、実行形式のプログラムの動作を管理するのに用いられ、ファンクションコール 4BH の実行時に、メモリエリアの先頭から 100H バイトの領域に割り当てられる。

00H			プロセス最終セグメント+1
04H	システム予約域		
08H			プロセス終了時の抜出し
0CH	アドレス		INT23Hルーチンの
10H	アドレス(コピー)		INT24Hルーチンの
14H	アドレス(コピー)		
18H			
	システム予約域		
28H			
2CH	プロセス環境セグメント		上位コントローラセグメント
30H			
	システム予約域		
4CH			
50H	INT 21H	RETF	
54H			
	システム予約域		
7CH			
80H			
	デフォルトのシステム情報エリア		
FCH			

7.3 システムコール

INT 21H により、MS-DOS と互換性のあるシステムコールが実現されている。これにより、ファイルアクセス／メモリ管理を簡単に行うことができる。

システムコールの一般型は、

- 入力 AH：機能番号
その他のレジスタ：引数（もしあれば）
- 出力 CF= 1：エラー
AX：エラー番号
CF= 0：正常終了
その他のレジスタ：出力（もしあれば）

であり、フラグと入出力に使用されるレジスタ以外のレジスタは保持される。

●システムコール一覧

機能番号 (16進)	機 能
0E	デフォルト・ドライブの選択
19	デフォルト・ドライブの取得
1A	システム情報エリアの取得
1B	デフォルト・ドライブ・データの取得
1C	ドライブ・データの取得
25	割込みベクタの設定
2E	リードアフタライトモードのセット／リセット
2F	システム情報エリアのアドレスの取得
3A	プロセスの常駐終了
33	CTRL-Cの検査状態の設定
35	割込みベクタの取得
36	ディスクのフリースペースの取得
39	ディレクトリの作成
3A	ディレクトリの削除
3B	カレント・ディレクトリの変更
3C	ファイルの作成
3D	ファイルハンドルのオープン
3E	ファイルハンドルのクローズ
3F	ファイルデータの読み込み
40	ファイルデータの書き込み
41	ファイルの削除
42	ファイルポインタの移動
43	ファイル属性の取得／設定
45	ファイルハンドルの二重化
46	ファイルハンドルの強制二重化
47	カレント・ディレクトリパスの取得
48	メモリエリアの割当て
49	割り当てたメモリエリアの解放
4A	割り当てたメモリエリアの変更
4B	プロセスの実行
4C	プロセスの終了
4D	下位プロセスから出力コードの取得
4E	最初に一致するファイルの検索
4F	次に一致するファイルの検索
54	リードアフタライトモードの取得
56	ディレクトリエントリの変更
57	ファイルの日付／時刻の取得／設定
58	メモリ割当てモードの取得／設定
5A	一時ファイルの自動作成
5B	新たなファイルの作成

0EH	デフォルトドライブの選択
-----	--------------

入 力 AH=0EH

AL=デフォルトドライブ番号 (A:0, B:1...)

出 力 CF=エラーフラグ

AL=接続ドライブ数 (正常終了時)

解 説 指定されたドライブをデフォルトドライブとして設定する。この設定は、パス名のドライブ指定を省略した場合などに有効となる。また現在接続されているドライブの総数を AL に返す。

19H	デフォルトドライブの取得
-----	--------------

入 力 AH=19H

出 力 AL=デフォルトドライブ番号 (A:0, B:1...)

解 説 デフォルトドライブとして選択されているドライブ番号を AL に返す。

1AH	システム情報エリアの設定
-----	--------------

入 力 AH=1AH

DS:DX=システム情報エリアの開始アドレス

出 力 なし

解 説 システム情報エリアの開始アドレスを設定する。このエリアは、ファンクション 4EH, 4FH でファイルの検索を行う時などに有効である。デフォルトの位置としては、プロセスコントローラのオフセット 80H が用いられる。このエリアにデータを転送する場合、システムは上で設定した DS の値をそのまま用いるため、DX にはラップ・アラウンドを起こさない範囲の値を指定する必要がある。現在のシステム情報エリアのアドレスは、ファンクション 2FH を用いて取得することができる。

1BH	デフォルトドライブデータの取得
-----	-----------------

入 力 AH=1BH

出 力 AL=1 クラスタ当たりのセクタ数

CX=1 セクタ当たりのバイト数

DX=1 ドライブ当たりのクラスタ数

DS:BX=メディア ID の格納領域アドレス

解 説 デフォルトドライブの物理情報を各レジスタに返す。またメディア ID を格納しているエリアのアドレスを DS:DX に返す。類似のファンクションとして、ファンクション 1CH, 36H がある。

〈メディア ID とディスクフォーマット〉

ID	タイプ	セクタ長	セクタ/トラック	ディスク容量
01H	2D	256 バイト	16 セクタ	320KB
02H	2HD	256 バイト	26 セクタ	1MB
F9H	2DD	512 バイト	9 セクタ	720KB
FBH	2DD	512 バイト	8 セクタ	640KB
FDH	2D	512 バイト	9 セクタ	360KB
FEH	2HD	1024 バイト	8 セクタ	1.2MB
FFH	2D	512 バイト	8 セクタ	320KB
D0H	HD	—	—	5MB
D1H	HD	—	—	10MB
D2H	HD	—	—	20MB
D3H	HD	—	—	40MB

1CH	ドライブデータの取得
-----	------------

入 力 AH=1CH
DL=ドライブ (デフォルト: 0, A: 1, B: 2, ...)

出 力 AL=FFH: 無効なドライブ
 *FFH: 1 クラスタ当たりのバイト数
CX=1 セクタ当たりのクラスタ数
DX=1 ドライブ当たりのクラスタ数
DS:BX=メディア ID の格納領域アドレス

解 説 指定されたドライブの物理情報を各レジスタに返す。指定されたドライブが存在しない場合、AL に FFH を返す。この時 AL 以外のレジスタは意味を持たない。指定された番号のドライブが存在する場合、各レジスタにドライブの物理情報を返す。またメディア ID を格納しているエリアのアドレスを DS:BX に返す。類似のファンクションとして、ファンクション 1BH, 36H がある。

25H	割込みベクタの設定
-----	-----------

入 力 AH=25H
AL=割込みタイプ番号
DS:DX=割込み処理ルーチンの開始アドレス

出 力 なし

解 説 指定された割込みベクタの内容を変更する。通常、割込みベクタを変更する場合には、そのベクタがすでに使用されているかどうかをチェックする必要がある。もし使用されている場合、ユーザはまずファンクション 35H により旧割込みベクタの値を読み出し、それが新たな割込み処理ルーチンの出口となるように、ルーチンを組む必要がある。

2EH	リードアフタライトモードのセット／リセット
-----	-----------------------

入 力 AH=2EH

AL=00H：リードアフタライトの解除

01H：リードアフタライトの設定

出 力 なし

解 説 リードアフタライトモードの設定または解除を行う。リードアフタライトが設定された場合、ディスクヘータを書き込んだ後、チェックリードにより結果が確認される。ディスクに重要なデータを書き込む際に信頼性を向上させるのに有効だが、処理時間が2倍近くになるため、使用時には使い分けが必要となる。またデフォルトはオフ状態に設定されている。

2FH	システム情報エリアのアドレスの取得
-----	-------------------

入 力 AH=2FH

出 力 ES:BX=システム情報エリアの開始アドレス

解 説 ファンクション 1AH で設定したシステム情報エリアのアドレス返す。プロセス実行開始後、一度もファンクション 1AH が実行されていない場合、デフォルト値としてプロセスコントローラのオフセット 80H のアドレスを返す。

30H	プロセスの常駐終了
-----	-----------

入 力 AH=30H

AL=プロセス出力コード

DX=常駐するメモリのパラグラフサイズ

出 力 戻らない（親プロセスに戻っていく）

解 説 メモリを解放せずにプロセスを終了する。DX には在駐するメモリエリアのパラグラフサイズが、また AL にはプロセスの出力コードが設定されていなければならない。EXE 形式のファイルの場合、一般にプログラムとデータは同一セグメントに含まれるとは限らないため、DX にはデータ領域を含むプログラム全てのサイズを指定する必要がある。またプログラムの直前のプロセスコントローラ大きさ (100H) を含んでいる必要もある。このファンクションは現在のプロセスを終了し、プロセス ID (PID) のメモリサイズを DX の値に再設定する。このとき、終了プロセスに関連する他のメモリアサインには手を付けない。

また AL の出力コードは上位プロセスに情報を引き渡すためのものであり，親プロセスはファンクション 4DH を通して読み出すことができる。

33H	CRTL-C の検査状態の設定／取得
-----	---------------------------

入 力 AH = 33H
 AL = 00 : CTRL-C の検査状態の取得
 = 01 : CTRL-C の検査状態の設定
 (AL=01 のとき)
 DL = 00 : CTRL-C の検査を禁止
 = 01 : CTRL-C の検査を許可

出 力 AL = FFH : AL の指定が無効
 ≠ FFH : 正常終了
 (設定状態取得のとき)
 DL = 00H : CTRL-C の検査は禁止
 = 01H : CTRL-C の検査は許可

解 説 INT 21H システムコールの各ファンクションにおいて，CTRL-C の検査を行うかどうかを指定する。CTRL-C の検査が許可されている状態で，その入力が発出された場合，システムは INT 23H を実行する。

35H	割込みベクタの取得
-----	------------------

入 力 AH = 35H
 AL = 割込みタイプ番号

出 力 ES : BX = 割込み処理ルーチンの開始アドレス

解 説 指定した割込みベクタのアドレスを読み出す。このファンクションはファンクション 25H と組になっており，通常はベクタを変更する場合，その処理ルーチンの拔出しアドレスは，前もってこのファンクションで読み出しておいたアドレスに設定すべきである。

36H	ディスクのフリースペースの取得
-----	------------------------

入 力 AH = 36H
 DL = ドライブ番号 (デフォルト : 0, A : 1, B : 2)

出 力 BX = 使用可能なクラスタ数
 DX = 1 ドライブ当たりのクラスタ数
 AX = FFFFH : 無効なドライブ番号
 ≠ FFFFH : 1 クラスタ当たりのセクタ数

解 説 指定したドライブの使用可能なクラスタ数を返す。ドライブ番号が無効の場合，AX に FFFFH が入る。

39H	ディレクトリの作成
-----	-----------

入 力 AH=39H
DS:DX=パス名の先頭アドレス

出 力 CF =0: 正常終了
 =1: エラー
 AX=03H: 無効なパス名
 =05H: アクセス不可
 =0FH: 無効なドライブ

解 説 DS:DX によって指定された ASCIZ 形式のパス名を解析し、対応するサブディレクトリを作成する。パス中に無効な文字が検出された場合、解析を中断して AX にエラー 03H を返す。またパス名のドライブ指定が無効な場合、AX にエラー 0FH を返す。

3AH	ディレクトリの削除
-----	-----------

入 力 AH=3AH
DS:DX=パス名の先頭アドレス

出 力 CF =0: 正常終了
 =1: エラー
 AX=03H: 無効なパス名
 =05H: アクセス不可
 =0FH: 無効なドライブ
 =10H: 現在使用中のディレクトリ

解 説 指定されたサブディレクトリを削除する。DS:DX は削除するサブディレクトリのパス名を示す ASCIZ 形式の文字列を指していなければならない。削除するパス名が存在しない場合、AX にエラー 03H を返す。また削除するパス名がルートディレクトリであるか、ディレクトリが空でないか、属性がリードオンリーである場合、アクセスを否定して AX にエラー 05H を返す。さらに指定されたパスの指すディレクトリがカレント・ディレクトリである場合、AX に 10H を返す。
指定されたパスが正常な場合、そのサブディレクトリに対応するディレクトリエントリ、および FAT 上のクラスタチェーンを解放する。

3BH	カレントディレクトリの変更
-----	---------------

入 力 AH=3BH
DS:DX=パス名の先頭アドレス

出 力 CF =0: 正常終了
 =1: エラー

AX=03H：無効なパス名
 =05H：アクセス不可
 =0FH：無効なドライブ

解 説 DS：DX が指す ASCIZ 形式のパス名にカレントディレクトリを変更する。変更するパス名が存在しない場合、AX にエラー 03H を返す。
 パス名のルートからの文字列の長さ（ドライブ指定と最初の \backslash 、および最後のヌル文字を除いた値）は 61 バイト以内でなければならない。

3CH	ファイルの作成
-----	---------

入 力 AH =3CH
 CX =ファイルの属性
 DS：DX=パス名の先頭アドレス

出 力 CF =0：正常終了
 AX=ファイルハンドル
 =1：エラー
 AX=03H：無効なパス名
 =04H：オープン不可（ハンドルに空きがない）
 =05H：アクセス不可
 =0FH：無効なドライブ

解 説 ファイルを作成し、作成したファイルをオープンし、そのハンドルを返す。同一名のファイルが存在する場合、そのファイルは初期化される。このファンクションを用いて、サブディレクトリないしボリュームラベルを作成することはできない。

〈ファイルの属性〉
 bit 0：リードオンリー属性
 bit 1：隠れファイル属性
 bit 2：システムファイル属性
 bit 5：保存ファイル属性

3DH	ファイルのオープン
-----	-----------

入 力 AH =3DH
 AL =オープン属性
 00：読出し
 01：書込み
 02：読出し／書込み
 DS：DX=パス名の先頭アドレス

出 力 CF =0：正常終了
 AX=ファイルハンドル

=1: エラー

AX=02H: 存在しないファイル

=03H: 無効なパス名

=04H: オープン不可 (ハンドルに空きがない)

=05H: アクセス不可

=0FH: 無効なドライブ

解説 このファンクションは DS:DX で指定されたファイルをオープンする。システムファイル、隠れファイルもオープン/アクセスすることができる。

3EH	ファイルのクローズ
-----	-----------

入力 AH = 3EH

DX = ファイルハンドル

出力 CF = 0: 正常終了

= 1: エラー

AX = 05H: アクセス不可

= 06H: 無効なハンドル

解説 ファンクション 3CH, 3DH によってオープンされたファイルをクローズする。ファイルハンドルに誤りがある場合、もしくはハンドルが既にクローズされている場合、AX にエラー 06H を返す。

3FH	ファイルデータの読出し
-----	-------------

入力 AH = 3FH

BX = ファイルハンドル

CX = 読み出すバイト数

DS:DX = 読出しバッファのアドレス

出力 CF = 0: 正常終了

AX = 読み出されたバイト数

= 1: エラー

AX = 05H: アクセス不可

= 06H: 無効なハンドル (オープンされていない)

解説 ハンドルによって指定されたファイルからデータを読み出す。エラーがない場合、AX には読み込んだデータのバイト数を返す。このため、ファイルポインタが既に終点に達しているファイルを読み出そうとした場合、AX の値は 0 になる。また CX で指定された読出しバイト数がファイルの残りより大きい場合、AX には CX よりも小さな値を返す。したがって、CX で指定されたバイト数の全てがバッファ領域に転送されるとは限らない。

40H	ファイルデータの書込み
-----	-------------

入 力 AH = 40H
 BX = ファイルハンドル
 CX = 書き込むバイト数
 DS:DX = 書込みバッファのアドレス

出 力 CF = 0: 正常終了
 AX = 書き込まれたバイト数
 = 1: エラー
 AX = 05H: アクセス不可 (書込み禁止など)
 = 06H: 無効なハンドル (オープンされていない)

解 説 ハンドルによって指定されたファイルにデータを書き込む。エラーが無い場合、AX には書き込まれたバイト数を返す。したがって、ディスクの空き容量が少ない場合、AX は CX で指定したバイト数よりも小さな値を示すことがある。このとき CF はセットされないため、このファンクションの実行後は必ず AX の値を調べる必要がある。

41H	ファイルの削除
-----	---------

入 力 AH = 41H
 DS:DX = 削除するパス名のアドレス

出 力 CF = 0: 正常終了
 = 1: エラー
 AX = 02H: 存在しないファイル
 = 03H: 無効なパス名
 = 05H: アクセス不可
 = 06H: 無効なドライブ

解 説 DS:DX によって指定されたパス名のファイルを削除する。リードオンリーのファイルを削除する場合には、まずファンクション 43H により属性を変更した後、このファンクションを用いる。

42H	ファイルポインタの移動
-----	-------------

入 力 AH = 42H
 AL = 移動方法
 00H: ファイルの先頭からの絶対位置に移動する
 01H: 現在の位置にオフセットを加算した位置に移動する
 02H: ファイルの終りにオフセットを加算した位置に移動する
 BX = ファイルハンドル
 CX:DX = 移動するバイト数 (32 ビット数)

出 力 CF =0: 正常終了

DX:AX=新規のポインタ位置 (32 ビット数)

=1: エラー

AX =01H: 無効な移動方法

=06H: 無効なファイルハンドル

解 説 指定されたハンドルのファイルの読み込み/書き込みポインタを移動する。このポインタは 32 ビットの整数表現によるファイルのバイト位置を示す。ファンクション 3FH, 40H はこのポインタを参照してファイルの読み込み/書き込みを行うため、このポインタを制御することにより、ファイルの一部をスキップしたり、逆に戻って読み出し/書き込みを行うことができる。

結果として得られる DX:AX は実際のポイント位置 (ファイル先頭からの絶対位置) を示す。したがって、AL=1, CX:DX=0 でこのファンクションを実行することにより、現在のファイルポインタの位置を知ることができる。また移動方法として 02H を選んだ場合、ファイル内の位置を得るためには、CX:DX<0 となるように設定する必要がある。

43H	ファイルの属性の取得/設定
-----	---------------

入 力 AH =43H

AL =00H: ファイル属性の取得

=01H: ファイル属性の設定

CX =設定すべき属性 (設定のとき)

DS:DX=パス名を指す文字列のアドレス

出 力 CF =0: 正常終了

CX=ファイル属性 (取得のとき)

=1: エラー

AX=01H: AL の範囲が指定外

=02H: パス名がボリュームラベルを指定

=03H: 無効なパス名

=05H: アクセス不可

=0FH: 無効なドライブ

解 説 ファイルの属性の取得または設定をする。ディレクトリ属性およびボリュームラベル属性を変更することはできない。変更不可能な属性が指定されるか、変更不可能なファイルの属性を変更しようとした場合、AX にエラー 05H を返す。

45H	ファイルハンドルの二重化
-----	--------------

入 力 AH = 45H

BX = ファイルハンドル

出 力 CF = 0 : 正常終了

AX = 新しいファイルハンドル

= 1 : エラー

AX = 04H : オープン不可 (ハンドルの空きがない)

= 06H : 無効なハンドル

解 説 指定されたハンドルをもとに新たなハンドルを作成する。したがって、ファイルは新旧どちらのハンドルを使用してもアクセスすることができる。しかし、ファイルポインタは1つしかないため、双方のハンドルが兼用するので注意。

46H	ファイルハンドルの強制二重化
-----	----------------

入 力 AH = 46H

BX = 既存のファイルハンドル

CX = 新規のファイルハンドル

出 力 CF = 0 : 正常終了

= 1 : エラー

AX = 04H : オープン不可 (ハンドルが使用されている)

= 06H : 無効なハンドル

解 説 ハンドルを指定されたハンドルにコピーする。コピー先のハンドルが既にオープンされているハンドルである場合、AX にエラー 04H を返す。

47H	カレントディレクトリパスの取得
-----	-----------------

入 力 AH = 47H

DL = ドライブ番号 (デフォルト : 0, A : 1, B : 2, ...)

DS : SI = ディレクトリパスを読み込む領域のアドレス

出 力 CF = 0 : 正常終了

= 1 : エラー

AX = 0FH : 無効なドライブ

解 説 指定したドライブのカレントディレクトリのパス名を DS : SI で指定された領域に読み込む。この領域には 63 バイトの長さが必要である。パス名はルートディレクトリからの相対位置で表わした ASCIZ 形式の文字列になる (先頭の ¥ はない)。

48H	メモリエリアの割当て
-----	------------

入 力 AH = 48H
BX = 割り当てるメモリのパラグラフサイズ

出 力 CF = 0 : 正常終了
AX = 割り当てたメモリエリアの先頭セグメントアドレス
= 1 : エラー
AX = 07H : PID 異常
= 08H : メモリ不足
BX = 割当て可能なメモリサイズ

解 説 指定された大きさのメモリエリアを割り当てる。要求を満足するメモリが存在する場合、新たに PID (プロセス ID) が設定される。メモリが確保されない場合は、確保できうる最大のメモリサイズ (パラグラフ単位) を返す。メモリの空き領域を検索中、矛盾が生じるような PID が発見されると、AX にエラー 07H を返す。この時の BX の値は意味を持たない。またいくつかの空きメモリエリアが連続している場合、それらはまず単一のメモリエリアに再構成された後で割当て可能かどうかの判断が行われる。割り当てる空きメモリエリアの選択は、ファンクション 58H で指定されたメモリ割当てモードに基づいて行われる。

<メモリ割当てモード：ファンクション 58H>
00H : 最下位の割当て可能な空きメモリ領域に確保する
01H : 全メモリ中で割当て可能な最小の空きメモリ領域に確保する
02H : 最上位の割当て可能な空きメモリ領域に確保する

49H	メモリエリアの解放
-----	-----------

入 力 AH = 49H
ES = 解放するメモリエリアの先頭セグメントアドレス

出 力 CF = 0 : 正常終了
= 1 : エラー
AX = 07H : PID 異常

解 説 ファンクション 48H によって割り当てたメモリエリア単位でメモリを解放する。

4AH	メモリエリアのサイズ変更
-----	--------------

入 力 AH = 4AH
BX = メモリエリアのセグメントアドレス
ES = 変更したいメモリサイズ (パラグラフ単位)

出力 CF =0: 正常終了

=1: エラー

AX=07H: PID 異常

=08H: メモリ不足

BX =変更可能な最大メモリサイズ

解説 ファンクション 48H で割り当てたメモリエリアのサイズを変更する。現在のメモリサイズよりも BX の指定するサイズの方が小さい場合、新たな空き領域が作られる。逆に、BX の指定するサイズの方が大きい場合、現在のメモリエリアの後（上位）に連続する空きエリアを1つにまとめたのちに、その空きエリアが十分な大きさを持つかどうかを調べる。

4BH/00H	プログラムの実行
---------	-----------------

入力 AH =4BH

AL =00H

DS:DX=実行するファイルのパス名のアドレス

ES:BX=パラメータブロック位置のアドレス

出力 CF =0: 正常終了

=1: エラー

AX =01H: AL の指定が範囲外

=02H: 存在しないファイル

=03H: 無効なパス名

=05H: アクセス不可

=07H: PID 異常

=08H: メモリ不足

解説 指定されたパス名のファイル（プログラム）をロード実行する。パス名は 63 バイト以内でなければならない。

このファンクションをコールする上位プロセスは、下位プロセスをロードし実行するのに十分なメモリ領域を、ファンクション 4AH によりあらかじめ解放しておく必要がある。このファンクションは、指定されたパラメータブロックに基づいてプロセスコントローラを作成し、DS:DX で指定されたファイルをロード・実行する。また上位プロセスでオープンされていたハンドルの全てを、下位プロセスに引き渡す。

下位プロセスに渡される環境セグメントは個々の環境情報が ASCIZ 形式の文字列で形成されている。それぞれの文字列は連続して並べられ、環境全体の最後は 00H で終わる。したがって、環境セグメントの最後には 00H が2つ連続する。

〈パラメータブロック: 6 バイト〉

DW 渡される環境セグメントのアドレス

下位プロセスが使用する環境エリアのセグメントアドレス。上位プロセスの環境をそのまま下位プロセスに渡す場合（通常）は 0000H を指定する。

- DD 渡されるコマンド行の格納アドレス
 下位プロセスが使用するコマンド行の格納アドレス、コマンド行は第1バイト目がコマンドの文字数、第2バイト以降がコマンド文字列であり、全体で128バイト以下でなければならない。

4BH/03H	プログラムのオーバーレイロード
---------	------------------------

入 力 AH = 4BH
 AL = 03H
 DS : DX = ロードするファイルのパス名のアドレ
 ES : BX = パラメータブロック位置のアドレス

出 力 CF = 0 : 正常終了
 = 1 : エラー
 AX = 01H : AL の指定が範囲外
 = 02H : 存在しないファイル
 = 03H : 無効なパス名
 = 05H : アクセス不可
 = 08H : メモリ不足
 = 0FH : 無効なドライブ

解 説 指定されたパス名のファイル（プログラム）を自分のメモリエリア内にロードする。したがって、このファンクションはメモリの割当ては行わず、指定されたセグメントからプログラムをロードし、リロケート処理を行うだけである。また、プロセスのメモリエリアを越えてロードを行うことはできない。

〈パラメータブロック〉

DW ロードセグメント
 プログラムをロードするセグメントアドレス。
 DW リロケーション要素
 そのプログラムが実行されるセグメントアドレス。

4CH	プロセスの終了
-----	----------------

入 力 AH = 4CH
 AL = プロセス出力コード

出 力 戻らない（親プロセスに戻っていく）

解 説 現在のプロセスを終了し、上位プロセスに制御を戻す。この時、AL にはプロセスの出力コードを設定する。この出力コードは、上位プロセスに情報を引き渡すためのもので、ファンクション 4DH を通じて上位プロセスに取得される。また現在のプロセスのハンドルは全て無効となり、上位プロセスのハンドル状態が復帰する。

4DH	出力コード
-----	-------

入 力 AH = 4DH

出 力 AH = プロセス終了コード
AL = プロセス出力コード

解 説 ファンクション 31H, 4CH によって設定されたプロセス終了コードおよびプロセス出力コードを取得する。このファンクションは子プロセス終了後の1回しか返さない。

〈プロセス終了コード〉

- 00：正常終了
- 01：CTRL-C 終了
- 02：ハードウェアエラー終了
- 03：プログラム常駐のままの終了

4EH	最初に一致するファイルの検索
-----	----------------

入 力 AH = 4EH

CX = 検索する属性
DS:DX = 検索するパス名 (ASCIZ 形式) のアドレス

出 力 CF = 0：正常終了
 = 1：エラー
 AX = 03H：無効なパス名
 = 05H：アクセス不可
 = 0FH：無効なドライブ
 = 12H：ファイル検出不可

解 説 指定されたパス名に該当するファイルを検索する。パス名にはワイルドカード[?, *]が使用できる。最初に検索されたファイル情報がシステム情報エリアに格納される。検索の対象は CX によって指定されたものより属性が弱いファイルである。例えば、CX に 00H が設定された場合、通常のファイルとリードオンリーのファイルのみの検索になる。また 16H が設定された場合、通常のファイルの他に、隠しファイル、システムファイル、ディレクトリが検索される。全てのファイルを検索する場合、CX に 3FH を設定すること。
検索した結果、パス名に該当するファイルが存在しなかった場合、AX にエラー 12H を返す。

〈システム情報エリアに格納されるデータ〉

- 00-14H : 予約域
- 15H : 検索属性
- 16-19H : ファイルの時刻・日付
- 1A-1DH : ファイルのサイズ
- 1E-2AH : 検索したファイル名

4FH	次に一致するファイルの検索
-----	---------------

入 力 AH = 4FH

出 力 CF = 0 : 正常終了

= 1 : エラー

AX = 05H : アクセス不可

= 12H : ファイル検出不可

解 説 ファンクション 4EH によって検索されたファイルの次のファイルを検索し、そのファイル情報をシステム情報エリアに格納する。ファイルの検索は、このファンクションを繰り返すことによって行うことができる。該当するファイルを全て検索してしまった場合、AX に 12H を返す。

54H	リードアフタライトモードの取得
-----	-----------------

入 力 AH = 54H

出 力 AL = 現在のリードアフタライトモードの値

00H : リードアフタライトモードではない

01H : リードアフタライトモードである

解 説 ディスク書き込み時のリードアフタライト機能の動作状態を AL に返す。
(ファンクション 2EH 参照)

56H	ディレクトリエントリの変更
-----	---------------

入 力 AH = 56H

DS : DX = 既存ファイルのパス名の格納アドレス

ES : DI = 新規パス名の格納アドレス

出 力 CF = 0 : 正常終了

= 1 : エラー

AX = 02H : 存在しないファイル

= 03H : 無効なパス名

= 05H : アクセス不可

= 0FH : 無効なドライブ

= 11H : ドライブ間変更

解 説 ディレクトリエントリを変更することにより、ファイル名を変更する。変更先に該当するファイル名がすでに存在する場合、AX にエラー 05H を返す。また変更元と変更先のドライブの指定が異なる場合、AX にエラー 11H を返す。2つのサブディレクトリは同一でも異なっていて構わない。

57H	ファイルの日付／時刻の取得／設定
-----	------------------

入 力 AH = 57H

AL = 00H : 日付／時刻の取得

= 01H : 日付／時刻の設定

BX = ファイルハンドル

(設定のとき)

CX = 設定する時刻

DX = 設定する日付

出 力 CF = 0 : 正常終了

CX = ファイルの最終編集時刻 (取得のとき)

DX = ファイルの最終編集日付 (取得のとき)

= 1 : エラー

AX = 01H : AL の指定が範囲外

= 05H : アクセス不可

= 06H : 無効なハンドル

解 説 AL が 00H の時、ファイルハンドルが示すファイルの日付／時刻情報を返す。AL が 01H の時、ファイルハンドルが示すファイルの日付／時刻情報を仮変更する。この変更日時を実際にディスクに書き込むのは、そのファイルがクローズされる時点となる。また、BX はオープンされているファイルハンドルでなければならない。

〈日付／時刻の構成〉

○時刻

bit 15-11 : 時 (5 ビット)

bit 10-5 : 分 (6 ビット)

bit 4-0 : 秒 (5 ビット)

○日付

bit 15-9 : 年 (7 ビット)

bit 8-5 : 月 (4 ビット)

bit 4-0 : 日 (5 ビット)

58H	メモリ割当てモードの取得／設定
-----	-----------------

入 力 AH = 58H

AL = 00H : メモリ割当てモードの取得

= 01H : メモリ割当てモードの設定

(設定のとき)

BX = 00H : 下位アドレスから

= 01H : 最小ブロック

= 02H : 上位アドレスから

出力 CF = 0 : 正常終了
(取得のとき)

AX = 00H : 下位アドレスから
= 01H : 最小ブロック
= 02H : 上位アドレスから
= 1 : エラー
AX = 01H : AL の指定が範囲外

解説 メモリ割当てモードを設定／取得する。メモリ割当てモードは、メモリエリアの割当て（ファンクション 48H）時に、空きメモリ空間の選択方法を指定するものである。

〈メモリ割当てモード〉

- 00 : メモリエリアを確保する際、空き領域のできるだけ下位アドレスに割当てを行う。
つまり、空きメモリ領域の検索は下位アドレスから行われ、最初に発見された適合サイズの空き領域に割り当てられる。
- 01 : メモリエリアを確保する際、可能な最小の空き領域に割当てを行う。つまり、空きメモリ領域の検索は下位アドレスから最終アドレスまで行われ、その中で適合可能な最小の空きメモリ領域に割り当てられる。
- 02 : メモリエリアを確保する際、空き領域のできるだけ上位アドレスに割当てを行う。
つまり、空きメモリ領域の検索は下位アドレスから最終アドレスまで行われ、その中で適合可能な最上位のメモリ領域に割り当てられる。

5AH	一時ファイルの自動作成
-----	-------------

入力 AH = 5AH
CX = ファイルの属性
DS : DX = 作成するパス名の格納オフセットアドレス

出力 CF = 0 : 正常終了
AX = 一時ファイルのファイルハンドル
= 1 : エラー
AX = 02H : パス名がボリュームラベルを指定
= 03H : 無効なパス名
= 04H : オープン不可（ハンドルに空きがない）
= 05H : アクセス不可
= 0FH : 無効なドライブ

解説 指定したサブディレクトリに一時ファイルを作成する。このパス名領域は、入力時の ASCIZ 文字列の長さより 13 バイト余計に必要であり、この追加領域に一時ファイルのファイル名が格納される。この時、ファイル名には現在の時刻をもとにしたものが採用され、属性には CX によってされたものが用いられる。ただし、属性にディレクトリ、ボリュームラベルを指定することはできない。また、出力時のパス (DS : DX) は、サブディレクトリ名から一時ファイル名へと変化する。

重複した一時ファイルが発生した場合、新たなファイル名を生成し直し、重複がなくなるまでこれを繰り返す。このファンクションは一時ファイルの作成を行うだけであり、一時ファイルが不用になった場合、これを削除する必要がある。

DS:DX

ディレクトリ名	00	(13 バイトの追加領域)
---------	----	---------------

↓

ディレクトリ名	≠	一時ファイル名
---------	---	---------

5BH	新たなファイルの作成
-----	------------

入 力 AH =5BH

CX =ファイルの属性 (ファンクション 3CH と同じ)

DS:DX=パス名の先頭アドレス

出 力 CF =0:正常終了

AX=ファイルハンドル

=1:エラー

AX=02H:パス名がボリュームラベルを指定

=03H:無効なパス名

=04H:オープン不可 (ハンドルに空きがない)

=0FH:無効なドライブ

=50H:すでに存在するファイル

解 説 ファイルを作成・オープンし、そのハンドルを返す。同一名のファイルが存在しない場合、ディレクトリ上に新たに指定された属性のファイルを作成し、読出し/書込みのためにオープンする。ただし、指定されたファイルの属性がリードオンリーの場合、読出しのみのオープンを行う。また、同一名のファイルが既に存在する場合、AX にエラー 50H を返す。このファンクションは、ファイルが既に存在する場合にエラーとなる以外は、ファンクション 3CH と同じ動作をする。

第8章 N₈₈ V1/V2モード

V1/V2 モードは 88M/F シリーズをエミュレーションするためのモードであり，CPU だけでなく，メモリ構成・表示系がエミュレーションモードで動作する。

8.1 メモリ

CPU のメモリ空間 1M バイトの一部がエミュレーションモードになるが，その他のアドレスはシステムメモリエリアを除き V3 モードと同様にアクセスでき，CPU をネイティブモードで動作させればメモリ空間全部を使用することも可能。

Z80 エミュレーションモードにおける CPU の CS/DS レジスタは 1000H とする。

メモリシステムのネイティブ/エミュレーションの切り換えはポート 153H で行われる。

＜メモリレイアウト＞

- ROM エリア (V3 モードと同様)

FFFFF	ROM 10	ROM 11	禁 止			RESERV	RESERV
F0000	ROM 00	ROM 01	ROM 02	ROM 03	ROM 04	ROM 05	RESERV
E0000							

ス
ロ
ット
に
解
放

システムメモリエリア (88モードエリアを通じてのみアクセス可能)

- RAM エリア

9FFFF	RAM BANK #0 (ERAMページ3)	RAM BANK #1	RAM BANK #2	RAM BANK...
80000	ERAM ページ 2			
	ERAM ページ 1			
40000	ERAM ページ 0			
20000	88 MODE			
10000	RAM AREA			
00000				

— PC-8801-02N 互換拡張 RAM としてアクセス

— 88M/F エミュレーションメモリ空間
※このエリアに N88V3 モードにおけるメイン RAM/VRAM/ROM の一部がマッピングされ，88M/Fシリーズの 64K バイトのメモリ空間をエミュレートする。

8.2 テキスト表示

V1/V2 モードでは、VA が持つ 64K の TVRAM の一部を 88 モードエリアにマッピングし、IDP をエミュレーションモードで動作させてアトリビュートコードを変換しながら文字表示を行う。

8.2.1 TVRAM

V3 モードにおける TVRAM の内、次の 4K バイトが 88 エリアにマッピングされる。

V3 モード： A6000H～A6FFEH



V1/V2 モード：1F000H～1FFFFH

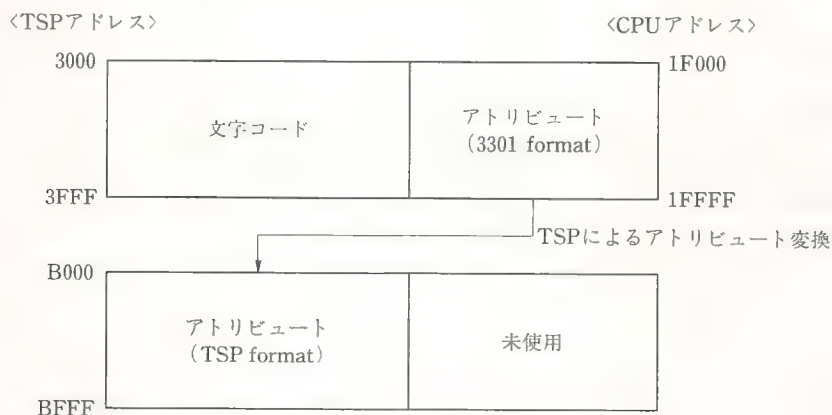
※1：A0000H～AFFFFH から TVRAM をアクセスすることはできない。

※2：DMA によるアクセスはできない。

このとき TSP はバイトアクセスモードに設定され、ローカルアドレスも変化する。

V3 モード：	3000H～37FFH (WORD)	3800H～3FFFFH (WORD)
	B000H～B7FFH (WORD)	B800H～BFFFFH (WORD)
	↓	↓
V1/V2 モード：	3000H～3FFFFH (BYTE)	使用不可
	B000H～BFFFFH (BYTE)	使用不可

マッピングされた TVRAM は 3301 の VRAM フォーマットとして使用され、この TVRAM の内容の表示を行うために、TSP はアトリビュートデータのフォーマット変換を自動的に行う。この場合カラーコード（8色）は、カラーコード 8～15 に変換される。



88 モードにおいても 3301 の以下の機能はエミュレーションされない。

- 1) 1行おき表示モード
- 2) ブロックカーソルのリバース指定
- 3) 特殊制御文字及び特殊制御文字による割込みの要求の発生
- 4) 水平カウンタのクリア
- 5) ライトペン機能（外部ハードウェアがサポートしていない）

- 6) アトリビュート無しモード
- 7) ノントランスペアレント・モノクロ・アトリビュートモード

8.2.2 キャラクタジェネレータ

V1/V2 モードの VRAM に対する表示のためのアクセスはバイト単位に行われる。また、文字のレターフェースは 8×8 ドット (15 KHzCRT)、8×16 ドット (24.8 KHzCRT) が標準となるため漢字表示は行えない。ボディフェースは横方向が 8 ドット固定で、縦方向が 8/10/16/20 ドットが設定できる。RAM キャラクタジェネレータ (外字) は使用できない。

漢字フォントについては、ポート E8H/E9H, ECH/EDH からアクセスできる漢字は M/F シリーズに搭載されている文字種だけであり、拡張文字/外字のアクセスはできない。

8.2.3 文字コード・アトリビュートコード

文字コードとして M/F シリーズと同じ 8 ビットコードが使用される。

アトリビュートモードとしてはトランスペアレントモード (カラー/モノクロ) のみがサポートされている。アトリビュートコードは M/F シリーズと同じであるが、ファームウェアで処理している関係上、桁指定の方法は BASIC で採用されている標準的な方法をとる必要がある。

また、セミグラフィックスアトリビュートを設定した場合、1 行前の設定が次行に対して保持されず、各行の先頭で必ずリセットされてしまう。このため、セミグラフィックスアトリビュートは各行ごとに設定しなおす必要がある。

その他、以下の制限事項がある。

- ・アトリビュートエリアを 0 クリアしてはならない。
- ・同一行中で、同一列アドレスに複数のアトリビュートを設定してはならない。
- ・同一行中で、複数の列アドレスに異なるアトリビュートを指定する場合には必ず列アドレスの小さい順に設定する。
- ・表示画面の先頭では機能指定/色指定を必ず行う。

これらの制限に該当した場合には、正常に表示されなかったり、テキスト VRAM に対するアクセスが異常に遅くなったりする。

8.2.4 画面分割・スクロール機能

4 個ある分割画面うちの 1 画面を用いて表示が行われ、他の 3 個の分割画面を使用することはできない。したがって、画面分割はできない。また、スクロール等の機能もサポートされない。

なお V1/V2 モードの表示画面として使用される分割画面に対する画面制御パラメータはあらかじめ V3 モードで設定を行っておく必要がある。

8.2.5 40 桁モード

V1/V2 モードでは TVRAM に格納されている偶数番目の文字フォントを 2 倍に拡大し、奇数番目の文字は表示しない。ただし、アトリビュートの列指定は 80 文字で計算されるため、偶数に設定する必要がある。

8.3 グラフィックス機能

グラフィックス表示モードとしてはマルチプレーンモードとシングルプレーンモードとがあるが、V1/V2 モード時には、マルチプレーンモードを選択する。

CRT 表示モードは、使用する CRT によりインターレースモード 0 またはインターレースモード 1 を設定して使用することができる。

また、グラフィックススクリーンはスクリーン 0 のみが可能である。

水平解像度は 640 ドットを設定し、垂直解像度については 15 KHz CRT を使用時には 200 ライン、24.8 KHz CRT 使用時には 400 ラインを設定する。

8.3.1 GVRAM

V1/V2 モードでは、各プレーンあたり 16K バイトの容量をもっており、CPU メモリ空間の 1C000H～1FFFFH 番地にマップされる。

〈V3 モード (マルチプレーン)〉	〈V1/V2 モード〉
プレーン 0 : A4000H～A7FFFH	→ プレーン 0 (OUT 5CH)
プレーン 1 : A4000H～A7FFFH	→ プレーン 1 (OUT 5DH)
プレーン 2 : A4000H～A7FFFH	→ プレーン 2 (OUT 5EH)
プレーン 3 : 使用しない。	

8.3.2 ピクセルモード

ピクセルモードとしては 1 ビット/ピクセル、4 ビット/ピクセルモードが選択可能である。ただし 4 ビット/ピクセルモードでのプレーン 3 の表示データは使用しないため画面スイッチによりマスクしておく必要がある。

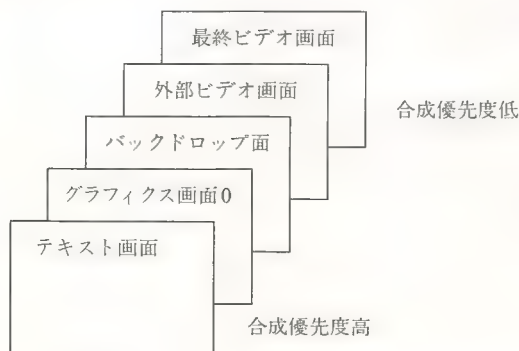
8.3.3 画面分割

表示画面はフレームバッファ 0 のみを使用するため、画面分割はできない。スクロールは可能である。

8.4 画面合成機能

CRT 画面上にはテキスト画面、グラフィック画面が合成されて表示される。スプライト機能は通常使用しないが、ハードウェア的には特に制限はない。

8.4.1 画面構成



9種の表示画面のうち、5種の表示画面によって合成された画面が表示される。ただしVAの表示機能の一部を使用しているだけであるため、その他の表示画面をイネーブルにして合成表示を行うことも可能である。

また、V1/V2モードにおいて、テキスト画面で使用されるカラーコードはカラー8～15の8色であるため、カラー判別レジスタには8を設定する。透明色の設定ではテキスト/スプライトのカラーコード0を除き、すべての色を不透明として設定する。画面マスク機能は使用しないように設定する。

バックドロップ面はバックドロップモード0で使用される。バックドロップカラーレジスタはM/Fシリーズのバックグラウンドカラーレジスタとは形式が異なるため、バックグラウンドレジスタに対するCPUのアクセスは、I/Oトラップによりソフトウェア的にエミュレーションされる。

8.4.2 カラーパレット

カラーパレット0はM/Fシリーズのパレット機能のエミュレーション用として使用し、カラーパレット1は8色ダイレクト指定のためのカラー変換テーブルとして使用する。

パレットの形式はM/Fシリーズと異なるため、M/Fシリーズ用パレットへのアクセスはI/Oトラップにより処理される。

また、パレットのプリンク機能は使用しない。

① デジタル/モノクログラフィックス パレットモード1

テキスト/グラフィックス合成画面…………… パレット1（8色固定パレット）

② デジタル/カラーグラフィックス パレットモード2

テキスト…………… パレット1（8色固定パレット）

グラフィックス…………… パレット0（8色中8色）

③ アナログ/モノクログラフィックス パレットモード0

テキスト/グラフィックス合成画面…………… パレット0（512色中8色）

④ アナログ/カラーグラフィックス パレットモード2

テキスト…………… パレット1（8色固定パレット）

グラフィックス…………… パレット0（512色中8色）

※1：カラーパレット0：パレット0～7を使用。

カラーパレット1：パレット8～15を使用。

※2：パレットモード2における指定画面レジスタには、
テキスト画面を設定して使用する必要がある。

索引

μ PD4990	10, 135
μ PD765	46
μ PD8241	9
μ PD8251	8, 19
μ PD8255	10, 46, 143
μ PD8255サブセット	28
μ PD8259	9, 128, 129, 130
μ PD9002	8, 12
16色モード	66
1 バイト文字	63, 114
1 バイト文字コード	113
2 バイト文字	63, 114
40文字表示	65
5 インチフロッピーディスク	142
8214互換ハードウェア	131
8214モード(V1/V2モード)	130
8214モード割込み制御	27
8241モード	128
8241モード割込みマスク	27
8259モード	128, 129
8ビットディップスイッチ	10
A	
ANK	9, 65, 116
B	
BEEP	10, 153
BEEP周波数発生器	133
C	
CGRAM	8
CGアドレスコード	114
CPU	12
CRT走査モード	117
D	
DMA機能	9
DISK BIOS作業領域	168

DMA	45, 54
DMA(ダイレクトメモリアクセス)機能	132
DMAインターフェイス	9
DMAコントローラ	39
DMAチャンネル	132
DMA転送機能	132
DMAモード	142, 146
DSH	93
DSP	93
DWIDダブルコントロール	63
F	
FBL	92
FBW	91
FDC強制レディの動作	147
FDC制御ポート 1	45
FDC制御ポート 2	45
FDCタイマ機能	147
FDCポート	46
FDCモードレジスタ	44
FDC割込み	129
FDD	9
FDDインターフェイス	142
FDDサブシステム	143
FDD制御インターフェイス	143
FDD用タイマ	134
FSA	91
G	
GVRAM	8, 49, 77, 84, 352
GVRAMステータス	25
GVRAM制御ポート 1	22
GVRAM制御ポート 2	23
GVRAM選択	25
GVRAMマップ	77, 84
H	
HDインターフェイス	147

I	
I/Oポートブロック	17
I/Oポートマップ	17
ICU	129, 131
ICU/8214互換ハードウェア	131
J	
JIS第1水準漢字	115
JIS第2水準漢字	115
JIS非漢字	115
N	
NEC非漢字	116
NMI	38
NTSC同期信号の切換え	121
O	
OCW1	129
OFX	92
OFY	93
P	
PC-Engine	11
PORT	10
R	
RAM	8
RAMエリア	47, 48
ROM	8
ROMエリア	47, 49
ROMバンク拡張ポート	50
ROMバンクステータス	38
RS-232C	8, 19
RS-232Cインターフェイス	148
RS-232C受信割込み	131
RS-232Cマスク	130
RS-232C割込み	129
S	
SGP	94
SGPコマンド	95
SGP作業領域の設定	102
SGPステータス	95
SGPパラメータ	95
SGP命令	95

SGP命令テーブル	102
SGP割込み	95, 129, 131
T	
TCU	42, 133
TCUカウンタ#0	133
TCUカウンタ#1	133
TCUカウンタ#2	133
TSP	35, 71
TSPアドレス	60
TVRAM	8, 49, 350
U	
UINT0(Bus#24)	129, 131
UINT1(Bus#25)	129, 131
UINT2(Bus#26)	129
UINT4(Bus#28)	129
USART	19
V	
V1/V2モード	8
V1/V2モード拡張RAM選択	26
V30モード	12
V3モード	8, 9, 47, 58
VA文字コード	112
VRTC割込み	129, 131
Y	
YM2203	24, 150, 152

ア	
アクセスブロックの選択	80
アクセスモード	79
アトリビュートコード	63
アナログVFO回路	143
アニメーションBIOS	252
アンダーライン	63
イ	
インターレースモード	121
インターフェイスサブCPU	140, 142
インターラプトベクタ	48
インテリジェントモード	142, 143

インテリジェントFDDインターフェイス.....28

ウ

ウェイト.....53

ウェイトサイクル.....53

エ

演算結果フラグ.....315

オ

オートスタートROM.....54

オペレーションコマンド.....141

SSG音源.....10

FM音源.....10

カ

外字.....9

外部ビデオ画面.....56

拡張ROM.....49

拡張ROM選択.....25

拡張アクセスモード.....80

拡張アクセス機能.....87

拡張グラフィックBIOS.....235

拡張システムコード.....48

拡張スロット.....11, 54

拡張ライト機能.....87, 87

拡張リード機能.....87

画面構成.....352

画面合成.....56, 352

画面合成機能.....352

画面分割.....88, 352

画面分割機能.....88

画面マスク機能.....108

画面マスクパラメータ.....35

画面マスクモードレジスタ.....32

カラー.....103

カラー階調.....103

カラーパレットモード.....105

カラーコード／ブリーンマスクレジスタ.....33

カラーパレット.....10, 353

カラーパレットモードレジスタ.....32

カレンダー時計.....10, 18, 135

カレンダー時計BIOS.....293

漢字.....9

漢字CGポート.....36

漢字ROM.....8, 49

キ

キーボードBIOS.....183

キーコードインターフェイス.....8

キーコード方式.....140

キーコードモード.....46

キーボード.....8

キーボードインターフェイス.....139

キーボードサブCPU.....140

キーボードサブCPU コマンドポート.....42

キーボードデータポート.....46

キーボードユニット.....142

キーボード割込み.....129

キーマトリクス.....18

キャラクタジェネレータ.....351

ク

グラフィック画面制御BIOS.....228

グラフィック画面制御パラメータ.....91

グラフィック画面制御レジスタ.....31

グラフィック機能.....76, 352

グラフィックス.....10

グラフィック表示解像度.....76

グラフィック表示禁止機能.....110

グラフィック画面0 透明色レジスタ.....34

グラフィック画面1 透明色レジスタ.....34

コ

固定ディスクユニットインターフェイス.....11

コマンドインターフェイス.....141

コミュニケーションBIOS.....263

混在文字コード.....112

サ

最終ビデオ画面.....56

サウンド.....10

サウンドBIOS.....273

サウンド機能.....150

サウンドコントローラ.....24, 129, 131, 150

サウンドコントローラ内タイマ.....134

サウンドコントローラ割込みマスク.....130

サブ画面.....88

シ

シークレット	63
辞書ROM	8, 49
システムインターラプトコール	323
システムエリア0	18
システムエリア1	30
システム起動プロセス	12
システム共通領域	48
システムコール	329
システム動作モード	8, 37
システムポート1	21
システムポート2	22
システムポート3	23
システムポート4	24
システムポート5	41
システムポート6, 7, 8	46
システムメモリエリア	47, 59
システムモードスイッチ	10, 125
実表示画面	122
実表示画面とフレームバッファ	122
シフトJISコード	113
受信バッファ	130
出力専用ポート保存領域	321
衝突グルーピングモード	70
衝突検出機能	69
衝突検出条件	70
衝突検出フラグ	70
シリアルコマンドモード	137
シングルプレーンモード	76, 84

ス

スーパーグラフィックプロセッサ	94
垂直解像度	77, 118, 121
スイッチ	10
水平解像度	77, 118, 122
水平ラップアラウンド	91
数値演算BIOS	314
数値表現	314
スキャナインターフェイス	10, 156
スキャナポート	26
スクリーンエディタBIOS	298

スクロール	88, 90
ストローブポート	23, 38
スピードスイッチ	126
スピードモード	127
スピードモードスイッチ	10
スプライト	9
スプライトBIOS	218
スプライト画面	57
スプライト制御テーブル	67
スプライトの解像度	69
スプライトのサイズ	68
スプライトの数と優先順位	68
スプライトの表示位置	69
スプライト表示機能	66
スプライトモード	66

タ

第一水準漢字フォント読出し	28
第二水準漢字フォント読出し	28
タイマ1 割込み	129
タイマカウンタユニット	42
タイマ機能	133
タイマ割込み	152
縦ズーム	69
ダブルウィドス	63

チ

調歩同期式	9
直接色指定画面	56
直接色指定画面制御レジスタ	31
直接色指定画面モード	86
直接色指定モード	86

テ

デジタルVFO回路	143
ディスクインターフェイス	142
ディスクサブシステム	143
ディスプレイ	10
ディップスイッチ	126
テキスト	9
テキスト/スプライト透明色レジスタ	34
テキスト/スプライトプロセッサ	71
テキストBIOS	200

テキストVRAM	59
テキスト画面	57
テキストコントローラ	35
テキスト制御ポート	36
テキスト表示	350
テキスト表示機能	58
テキスト表示禁止機能	66
スプライト表示機能	66
テキスト表示フレームバッファ	60
テキスト表示禁止機能	66
デバイスドライバ	323
ト	
同期信号マスク	111
同期モード	120
動作モード	142
透明色	107
独立アクセスモード	79
ドットアドレス	122
取込みフォーマット制御	121
ニ	
日本語入力フロントプロセッサBIOS	303
ネ	
ネイティブサービスルーチン	12
年機能使用モード	137
ノ	
ノンインターレースモード	118
ハ	
ハードウェアブロック図	7
ハードウェア文字コード	114
ハードディスクBIOS	173
ハードディスク(Bus#27)	129
ハードディスク制御	26
バックアップメモリ	52
バックドロップ画面	56
バックドロップカラーレジスタ	33
バックドロップ面	109
パラレルコマンド	135
パラレルコマンドモード	135
パレット	103, 104
パレット指定画面	55

パレット指定画面制御レジスタ	31
パレット指定画面モード	86
パレット指定モード	86
バンクRAM	48
ハンドシェイクポート	143
汎用タイマ	10, 133
汎用タイマ2	134
汎用タイマ2 割込み	131
汎用タイマ3	134
汎用タイマ3 (マウスタイマ)	129
汎用タイマ3 割込み	131
汎用タイマ3 制御ポート	44
ヒ	
比較読出し	80
ピクセル位置の指定	95
ピクセルサイズ	78, 84, 122
ピクセルモード	352
日付/時刻の設定	139
日付/時刻の設定・読出し	139
日付/時刻の読出し	139
ビットアサイン	87
ビデオサプレス機能	110
ビデオディジタルイズ	121, 122
ビデオボード	11
描画機能	10
表示解像度	117
表示画面制御レジスタ	30
表示機能	9
表示禁止機能	110
表示制御回路	58
標準RAM	8
フ	
ファイルアクセス	323
ファイルシステム	323
ファンシーフォントBIOS	296
ブリンク	63
ブリンク機能	106
プリンタ/カレンダー時計インターフェイス	18
プリンタBIOS	259
プリンタインターフェイス	8, 18, 148

フレームバッファ	89, 122
プレーンの選択	80
プロセスコントローラ	329
ブロック	96
フロッピーディスクBIOS	159

ホ

方形波発生	133
ボーレート発生器	133
ホリゾンタルライン	63

マ

マウスBIOS	285
マウスインターフェイス	154
マウスポート	11
マスクポート	38
マップ	15
マトリクススキュナインターフェイス	8
マトリクススキュン方式	139
マルチプレーンモード	76, 77

メ

メインCPU	140, 142
メインRAM	48
メインROM	8, 49
メモリ	47, 349
メモリアクセス	54
メモリオフセット	25
メモリオフセットインクリメント	26
メモリコントロール	50
メモリ常駐型ソフトスーパーバイザ	11
メモリスイッチ	127
メモリスイッチ書込み禁止/許可	42
メモリスイッチポート 1	20
メモリスイッチポート 2	21
メモリマップレジスタ	37, 50
メモリレイアウト	47, 349

モ

モードコマンド	141
モードスイッチ	125
モード設定	142
文字コード	63, 111

文字コード・アトリビュートコード	351
文字セット	111
文字フォント	111, 114
文字フォントアドレス	114
モノカラーモード	67

ユ

ユーザ定義文字	116
優先表示画面	57
優先順位設定	57

ラ

ラスタ	116
-----	-----

リ

リセット動作	142
リフレッシュモードレジスタ	38

ワ

割当て指定	57
割込み	127
割込み機能	9
割込みコントローラ	41, 130
割込みコントロール回路制御ポート	131
割込みテーブル	131
割込みベクタ	15, 130
割込みマスクポート	132
割込みモード変更	38
ワンショットタイマ	133

ご注意

- (1) 本書の一部又は全部について個人で使用するほかは、著作権上、(株)ビー・エヌ・エヌの承諾を得ずに無断で複写、複製することは禁じられています。
 - (2) 本書についての電話によるお問合せには一切応じられません。質問等がございましたら、往復はがき又は切手・返信用封筒を同封の上、弊社までお送り下さるようお願いいたします。
 - (3) 内容に関する責任は(株)ビー・エヌ・エヌにありますので、内容に関してメーカー等に直接問合わせることとは御遠慮下さい。
- 乱丁、落丁本はご面倒ですが弊社営業宛に御送付下さい。送料弊社負担にてお取替いたします。

PC-88VAテクニカルマニュアル

定価 3,500円

発行 1987年6月25日初版発行

著者 システムソフト監修
ビー・エヌ・エヌ第2企画部編

発行人 樺島正博

発行所 株式会社ビー・エヌ・エヌ
〒102 東京都千代田区麹町4-5 紀尾井町レジデンス5F
電話 営業部：03-238-1321 編集部：03-238-1322

装幀 NICK

印刷所 辻光舎印刷

©BNN corp. 1987 Printed in Japan

ISBN4-89369-024-8 C3055 ¥3500E

NECパーソナルコンピュータ **NEW**
PC-8800シリーズ
PC-88VA
テクニカル マニュアル

NEC
PC-8800
シリーズ

5
1
8
8

テ
ク
ニ
カ
ル
マ
テ
キ
ル

監修

システムソフト

BNN

第2企画部編

BNN

Bug News Network